

**Lab Manual- Deploy a sample app to your Google Kubernetes Engine (GKE)**

**Prepared for**:

**Date:** 18th May 2024

**Prepared by:**

Document Name: Lab Manual **Document Number** GCPLabn009

**Contributor:**

Contents

[1. Objective 3](#_Toc168482154)

[1. Single-zone versus multi-zonal 3](#_Toc168482155)

[2. Create a GKE Zonal Cluster 3](#_Toc168482156)

[3. Connect GKE Cluster with Kubectl 10](#_Toc168482157)

[1. In the Clsuter console click : extreme right and click connect 10](#_Toc168482158)

[2. Click Run in Cloud Shell 10](#_Toc168482159)

[3. Click following command in Cloud Shell 10](#_Toc168482160)

[4. Deploy and expose an application 11](#_Toc168482161)

# Objective

This Lab shows you how to create a Standard [zonal cluster](https://cloud.google.com/kubernetes-engine/docs/concepts/types-of-clusters#zonal_clusters) with the default features enabled in Google Kubernetes Engine (GKE). Zonal clusters have a single control plane in a single [zone](https://cloud.google.com/compute/docs/regions-zones). Depending on your availability requirements, you can choose to distribute your nodes for your zonal cluster in a single zone or in multiple zones.

**Important:** **Use regional clusters to run your production workloads as they offer higher availability than zonal clusters.**

## Single-zone versus multi-zonal

A **single-zone** cluster has a single control plane running in one zone. This control plane manages workloads on nodes running in the same zone. If you run a workload in a single zone, this workload is unavailable in the event of a zonal outage.

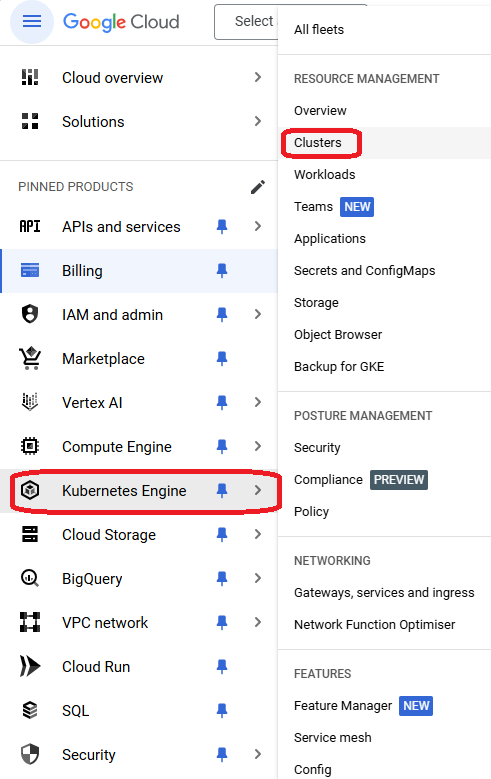
A **multi-zonal** cluster's nodes run in multiple zones, but it has only a single replica of the control plane. If you run a workload in multiple zones and there is a zonal outage, the workload is disrupted in that zone but remains available in other zones.

If you need higher availability for the control plane, consider creating a [regional cluster](https://cloud.google.com/kubernetes-engine/docs/concepts/types-of-clusters#regional_clusters) instead. In a regional cluster, the control plane is replicated across multiple zones in a region.

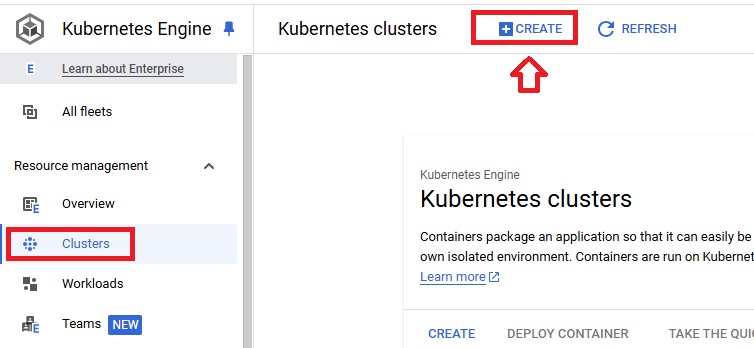
# Create a GKE Zonal Cluster

To create a zonal cluster with the Google Cloud console, perform the following tasks:

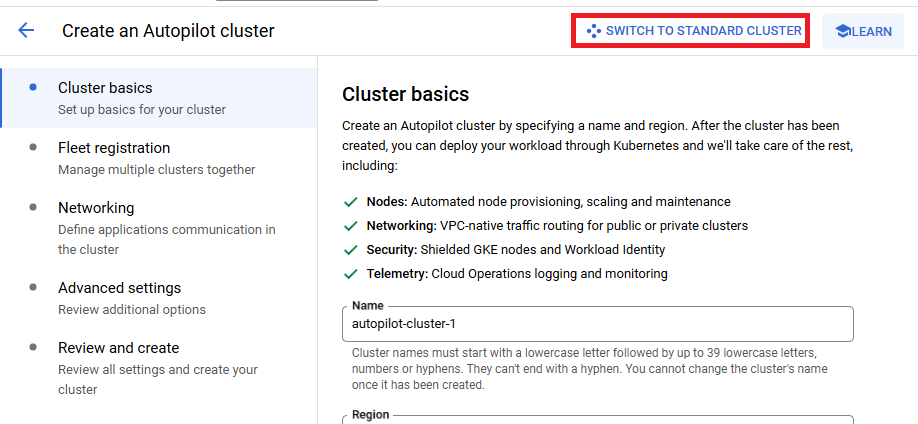
1. Go to the **Google Kubernetes Engine** page in the Google Cloud console.



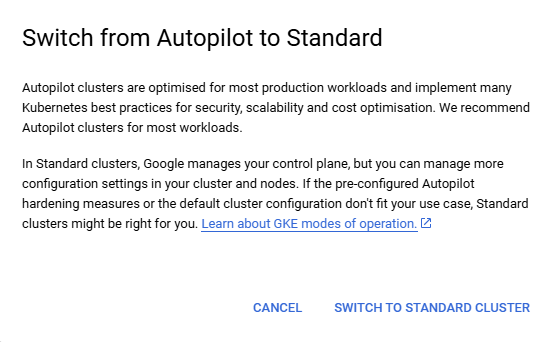
1. Click add\_box **Create**.



1. Click Switch to **Standard Cluster**

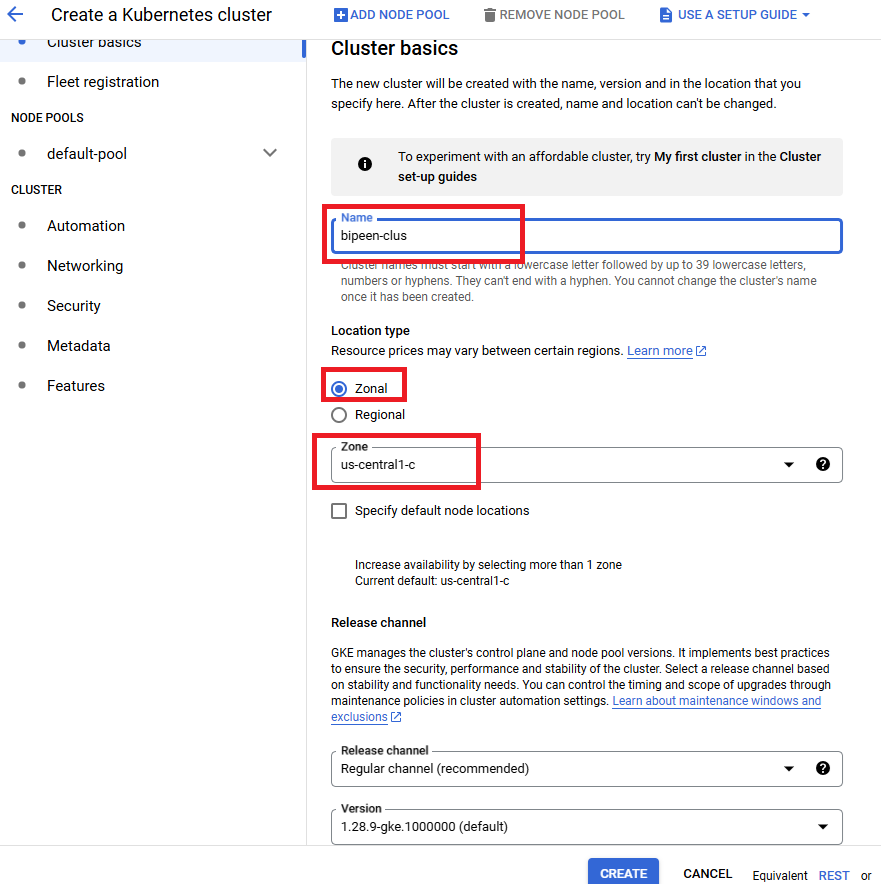


1. Click **Switch to Standard Cluster**

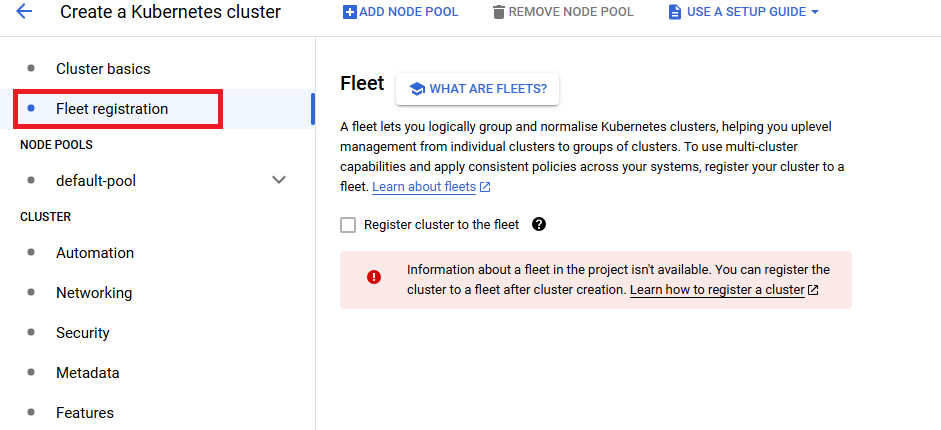


1. In the **Cluster basics** section, complete the following:

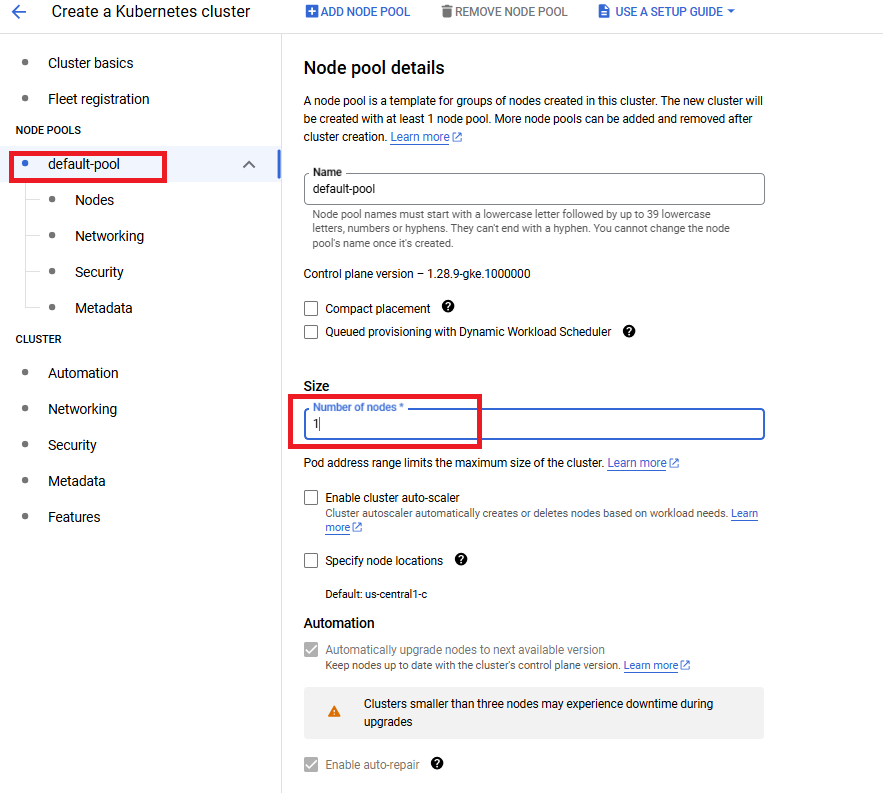
* Enter the **Name** for your cluster.
* For the **Location type**, select **Zonal**, and then select the [zone](https://cloud.google.com/compute/docs/regions-zones#available) for your cluster.



1. Leave all option default ( Optional (available with [GKE Enterprise](https://cloud.google.com/kubernetes-engine/enterprise/docs/concepts/gke-editions)): If you want to register your new cluster to a [fleet](https://cloud.google.com/kubernetes-engine/fleet-management/docs), go to the **Fleet registration** section, and follow the Google Cloud console instructions for [Create and register a new cluster](https://cloud.google.com/kubernetes-engine/fleet-management/docs/register/gke#console) to complete your cluster registration.



1. In The Node Pool, Select **Default pool** from left
   1. **Number of Nodes:** 1



1. In The Node Pool, Select **Nodes** from left

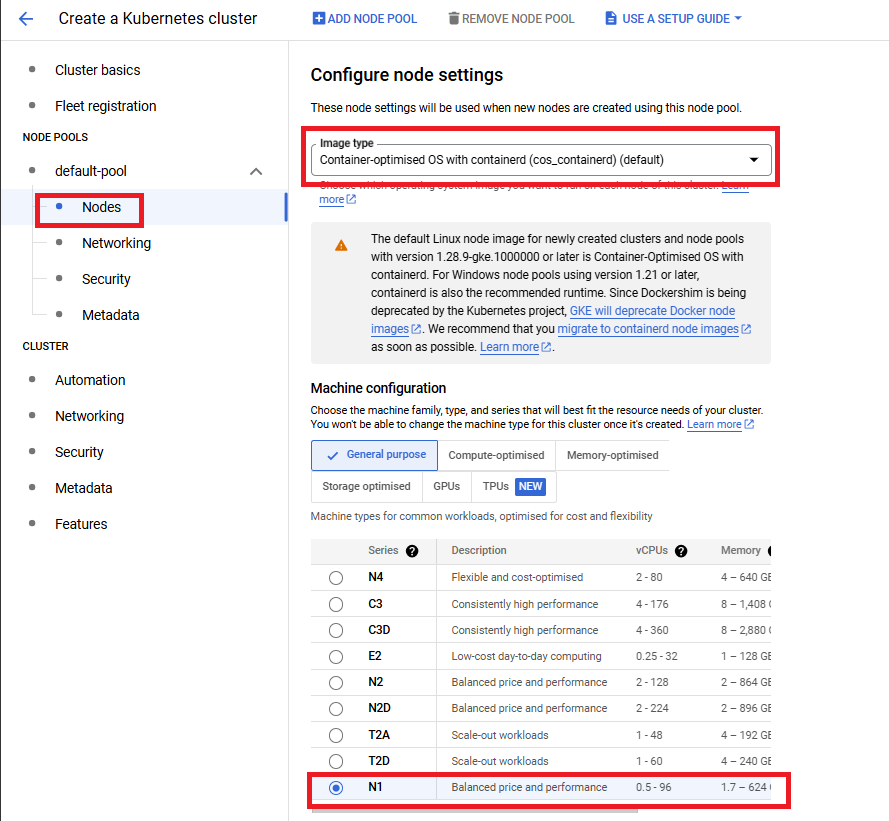
**Allowed Images:** - Container-optimised OS with containerd

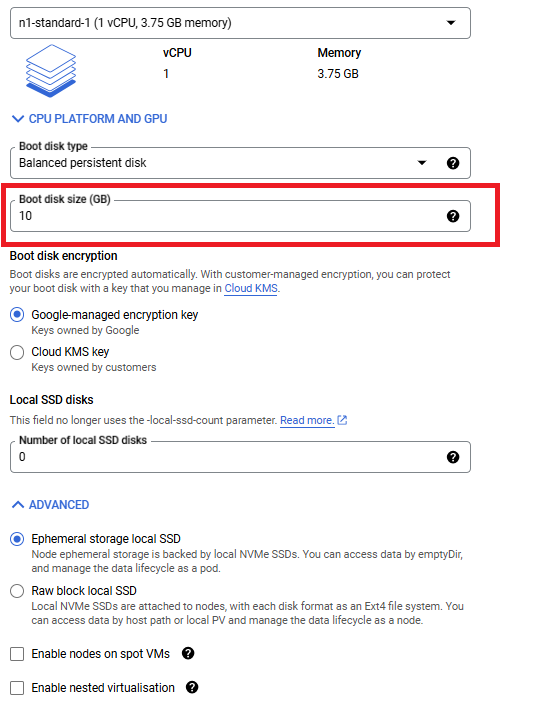
**VM Instance Type:** n1-standard-1

**Disk Size:** 10 GiB

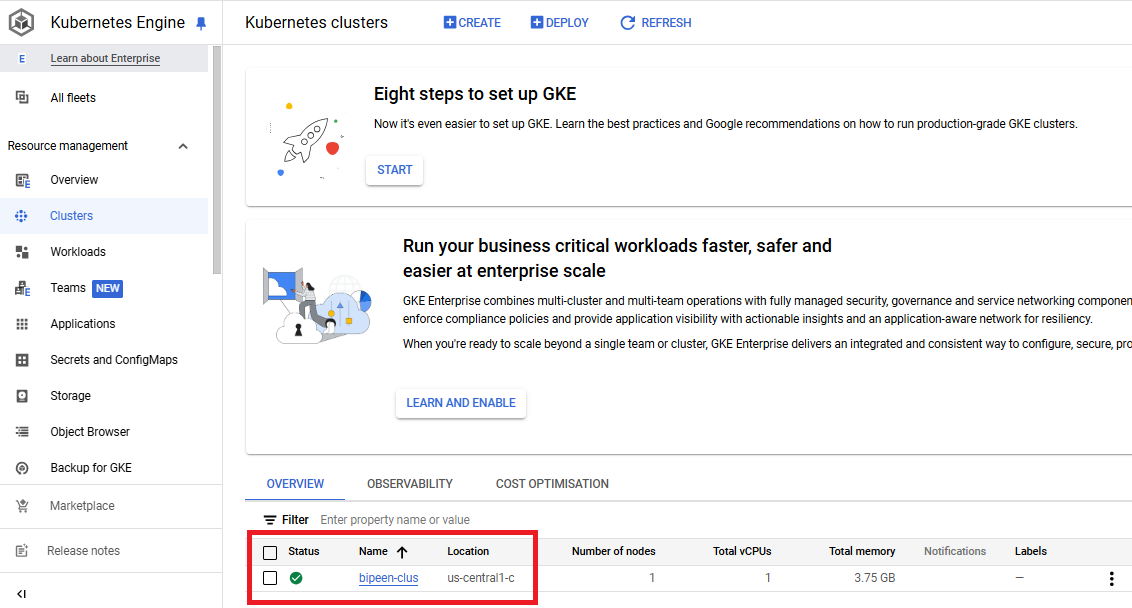
**Disk Types** - Balanced persistent disk

* 1. N1 as



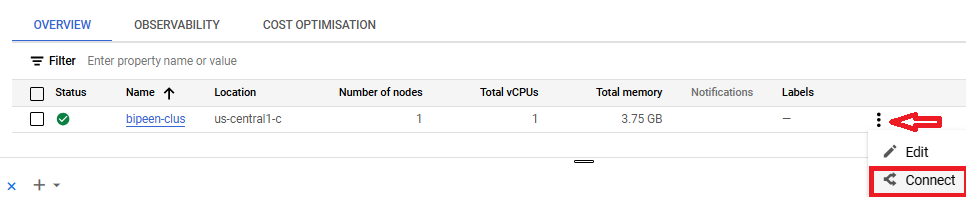


1. Click Create Button to Create the cluster
2. Once Installation Complete you see the Status Green

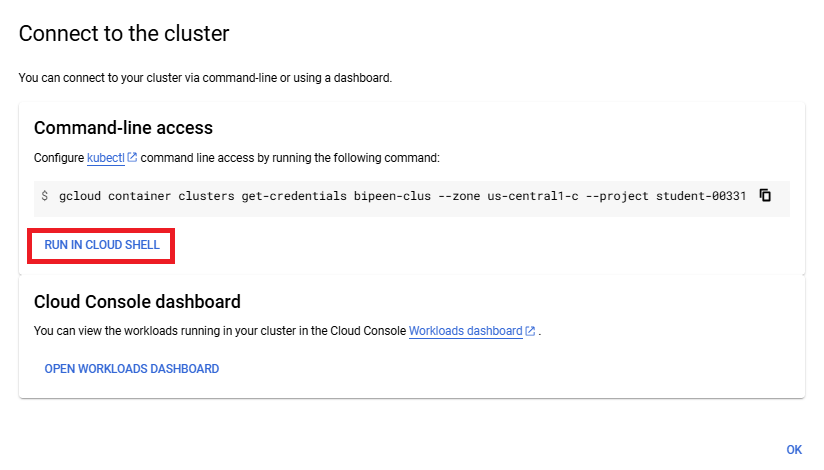


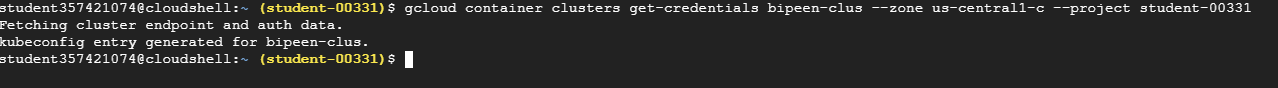
# Connect GKE Cluster with Kubectl

## In the Clsuter console click : extreme right and click connect



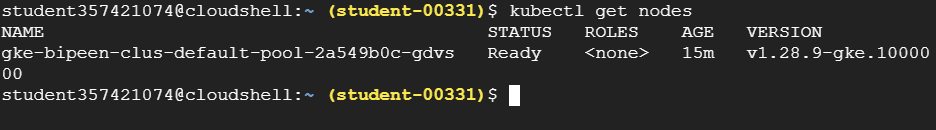
## Click Run in Cloud Shell





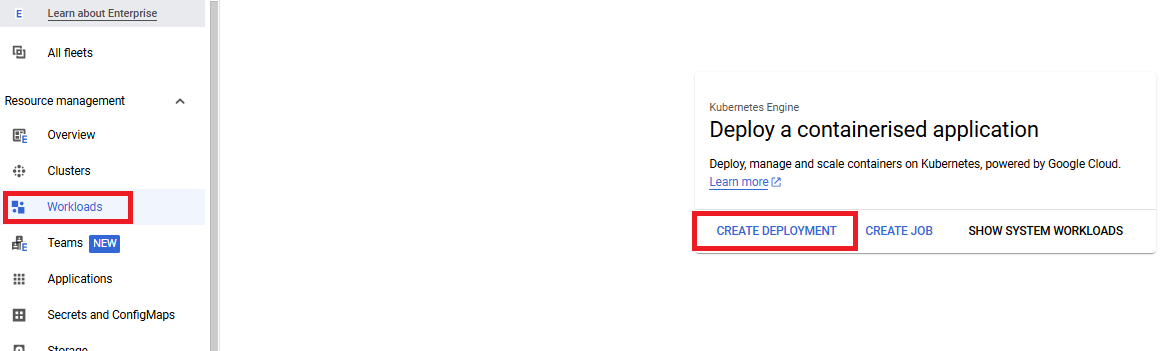
## Click following command in Cloud Shell

**kubectl get nodes**

****

# Deploy and expose an application

1. In the Google Cloud console, go to the GKE **Workloads** page.
2. Click **Deploy**.

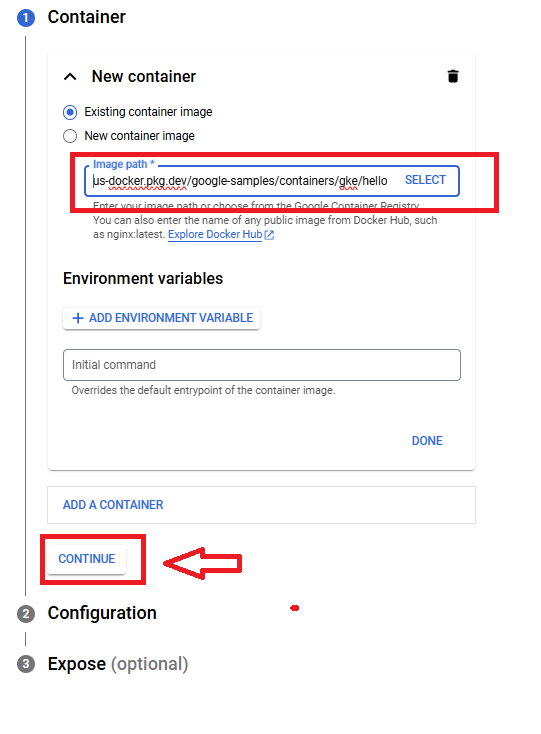


1. Leave **Existing container image** selected, and in **Image path** enter the following path:

**us-docker.pkg.dev/google-samples/containers/gke/hello-app:1.0**

This simple "hello world" app is packaged into a single container, but larger apps typically consist of several related containers that can be deployed together and run as a single workload.

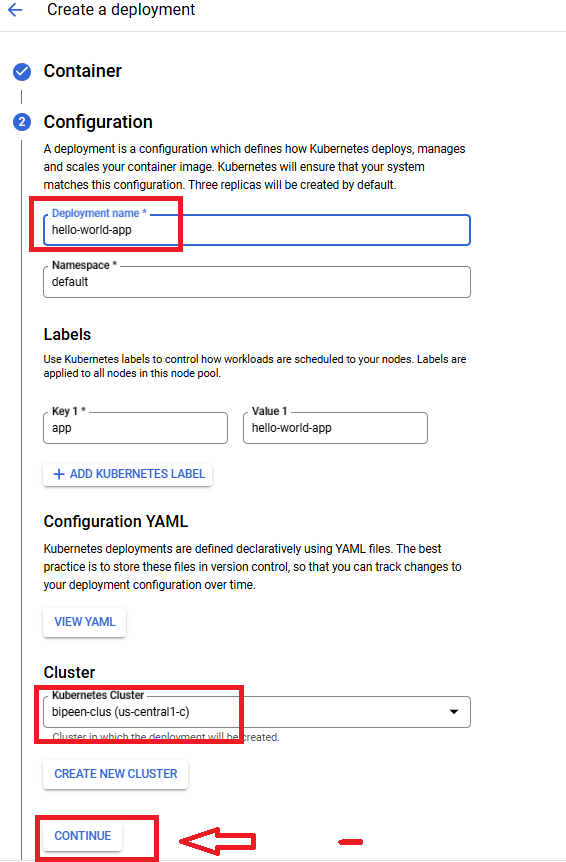
1. Click **Continue** to move to the **Configuration** section.



1. In **Deployment name**, enter the following name:

**hello-world-app**

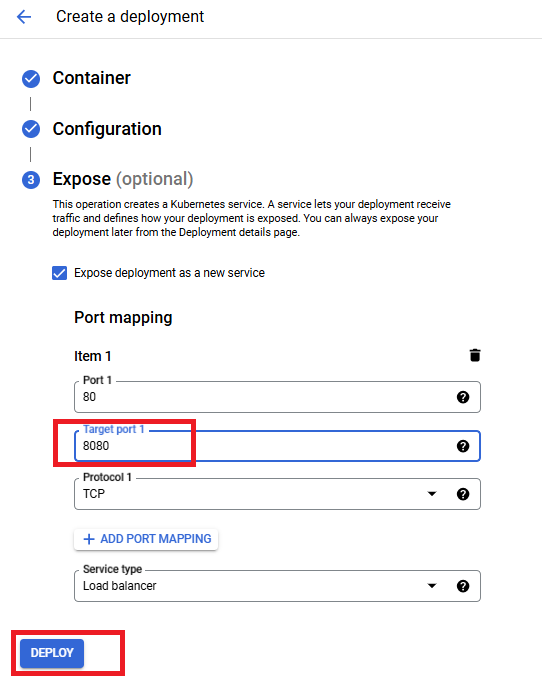
1. In Kubernetes Cluster, select **hello-world-cluster**.
2. Click **Continue**.



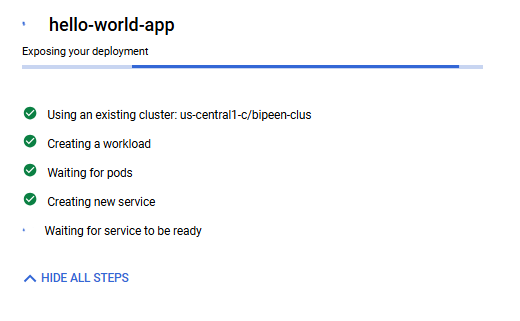
1. In the **Expose** section, create a load balancing Kubernetes Service to direct external requests to your app:
   1. Select **Expose deployment as a new service**.
   2. Leave **Port 1** set to **80**.
   3. In **Target port 1**, enter **8080**.
   4. Click **Deploy**.

GKE automatically assigns an available external IP address to the Service.

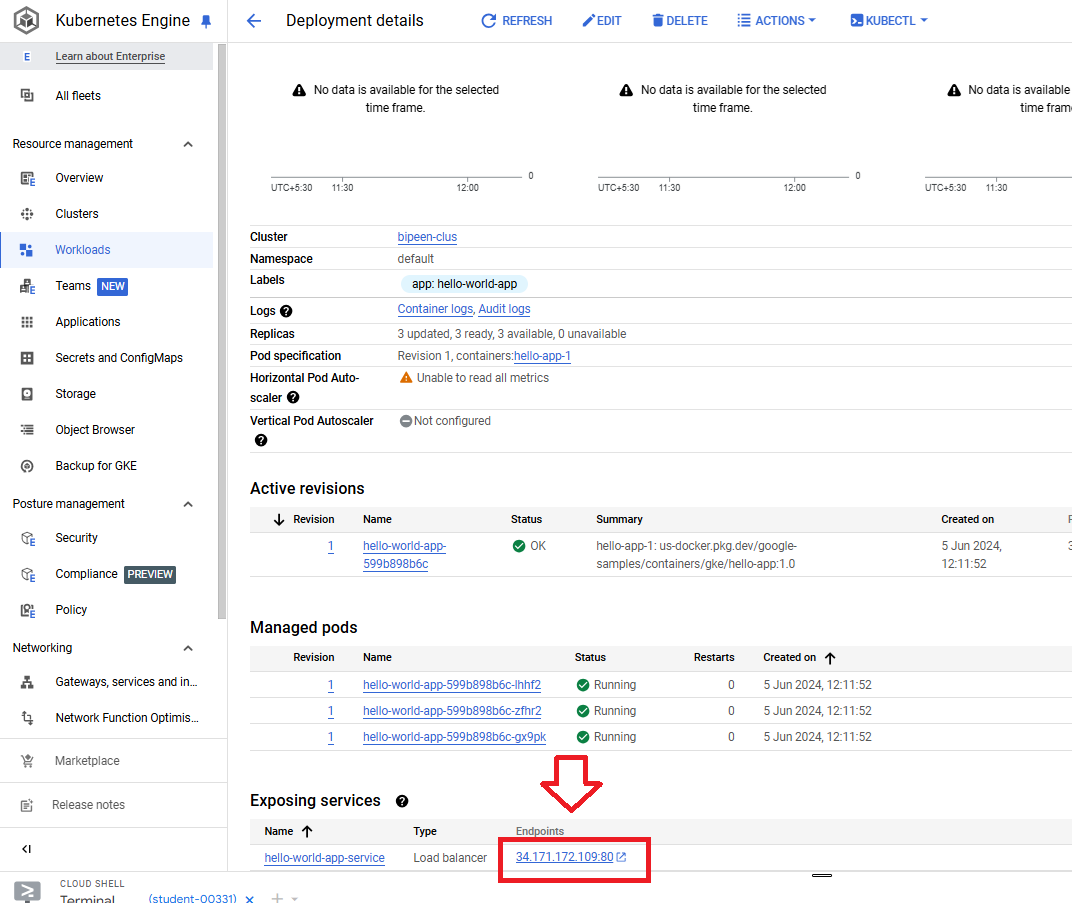
This Service is considered to be part of the hello-world-app workload.



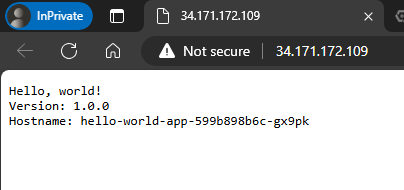
1. Wait until the deployment completes and you see the **Deployment details** page.



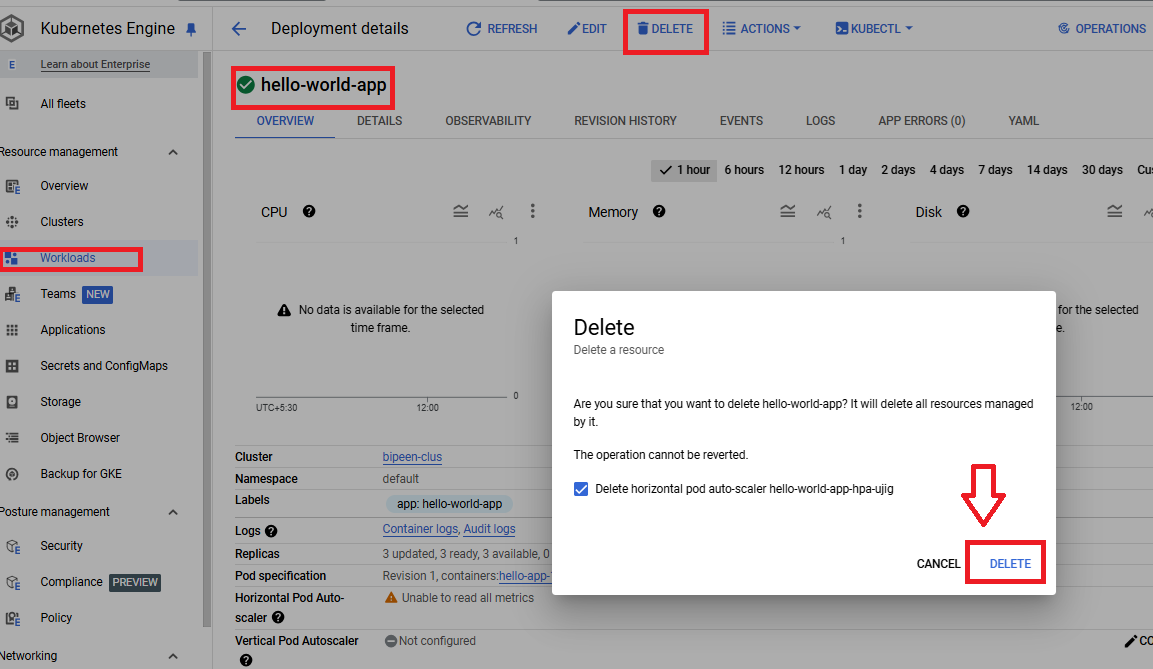
1. In the **Endpoints** column, click the **IP address**, which is publicly available.

****

GKE opens a new browser tab and sends a request to your app. Dismiss any secure-site warnings, and you should see **Hello, world!** in the new browser tab.

****

# Delete the deployment

****