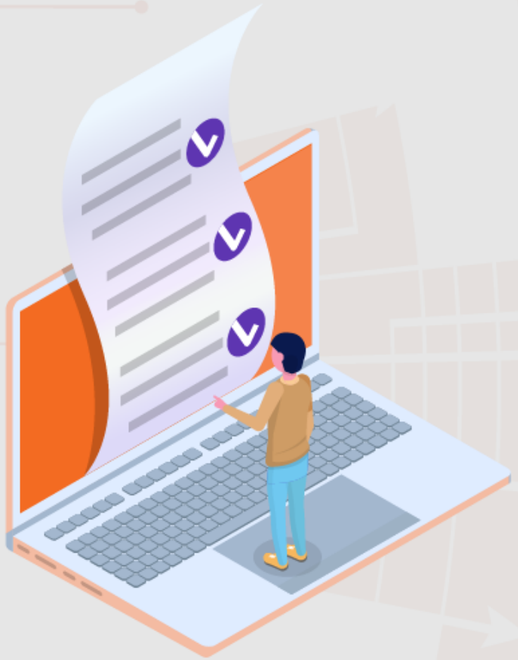
Container Orchestration using Kubernetes

Course-end Project

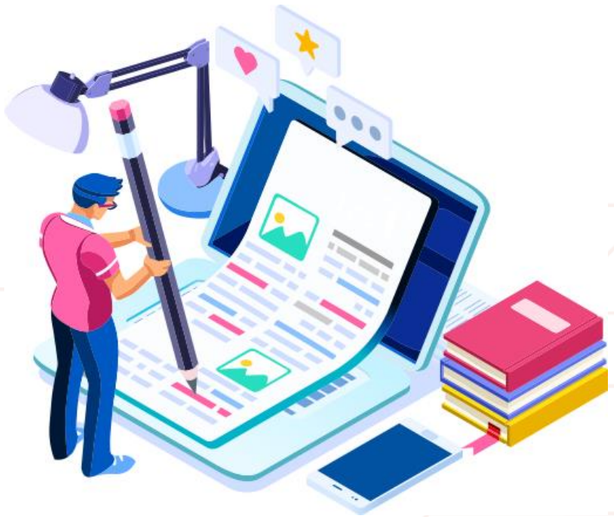
Deploy the Application Using the Kubernetes Dashboard

Objectives

To use the Kubernetes dashboard to deploy the application.

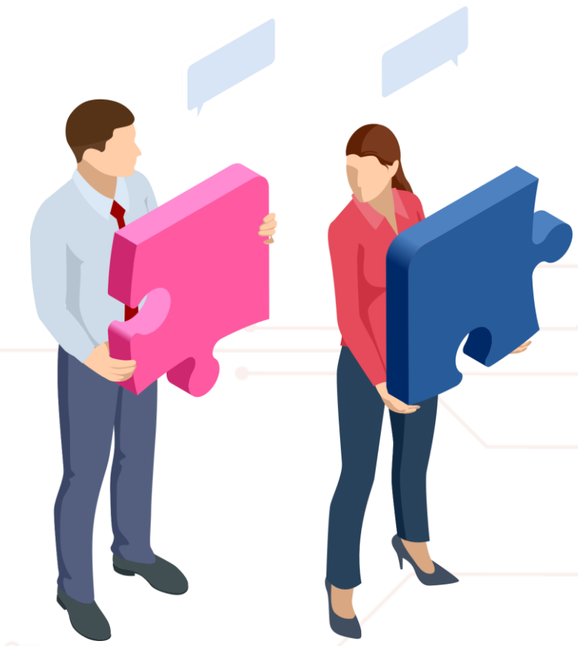


Prerequisites



- Kubeadm
- Kubectl
- Kubelet
- Docker

Problem Statement and Motivation



Problem Statement:

In this project, you should be able to create a multi-tier application using PHP and MySQL and deploy it using the Kubernetes dashboard with the Sandry service account.

Problem Statement and Motivation

Real-World Scenario:

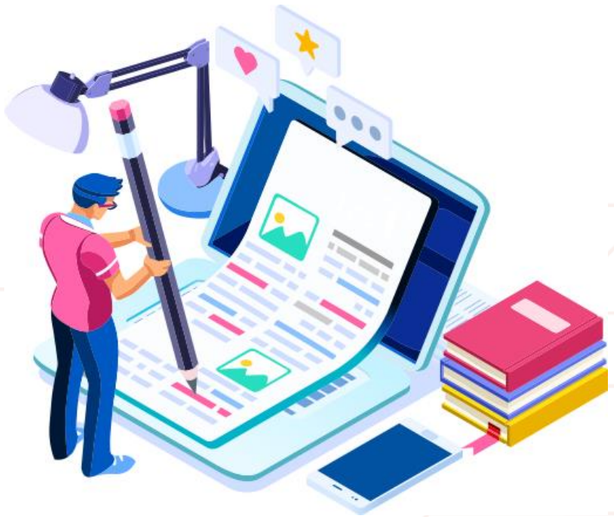
Your organization is looking to create a multi-tier application based on PHP and MySQL. Your job is to deploy this application using the Kubernetes dashboard. Create a user (service account) with the name of Sandry and make sure to assign her an admin role. WordPress and MySQL Pods should use Node3 as an NFS storage server using static volumes. The WordPress application must verify the MySQL service before getting it deployed. If the MySQL service is not present, then the WordPress Pod should not be deployed. These all should be restricted to the namespace called cep-project1 and must have 3 SVCs and 3 Pods as a max quota. All sensitive data should be using secrets and non-sensitive data should be using configmaps.

Industry Relevance

Skills used in the project and their usage in the industry are given below:

- **Kubeadm –** It is a tool that provides best practice "fast paths" for building Kubernetes clusters with kubeadm init and kubeadm join.
- **Kubectl -** It is just a command-line tool for running commands on Kubernetes clusters.
- **Kubelet -** It is the technology that applies, creates, updates, and destroys containers on a Kubernetes node.
- **Docker –** It delivers software as containers.

Task (Activities)



1. Getting started with Pods, Services, and Deployments
2. Creating and Verifying the Service
3. Creating a token and working on a dashboard
4. Configure the NFS-server for MySQL and WordPress Deployment
5. Setting up the NFS Client side
6. Creating and verifying the PV
7. Creating a secret for MySQL Deployments secret data
8. Creating a configmap for WordPress Deployment to store non-sensitive information

Project Reference

- **Task 1:** To deploy the Kubernetes dashboard, refer to the lesson 3; demo 6
- **Task 2:** To verify the Services, refer to the lesson 3; demo 6 and demo 7 to create an admin in the Kubernetes dashboard
- **Task 3:** To create a token for login into the Kubernetes dashboard using Sandry's secret key:

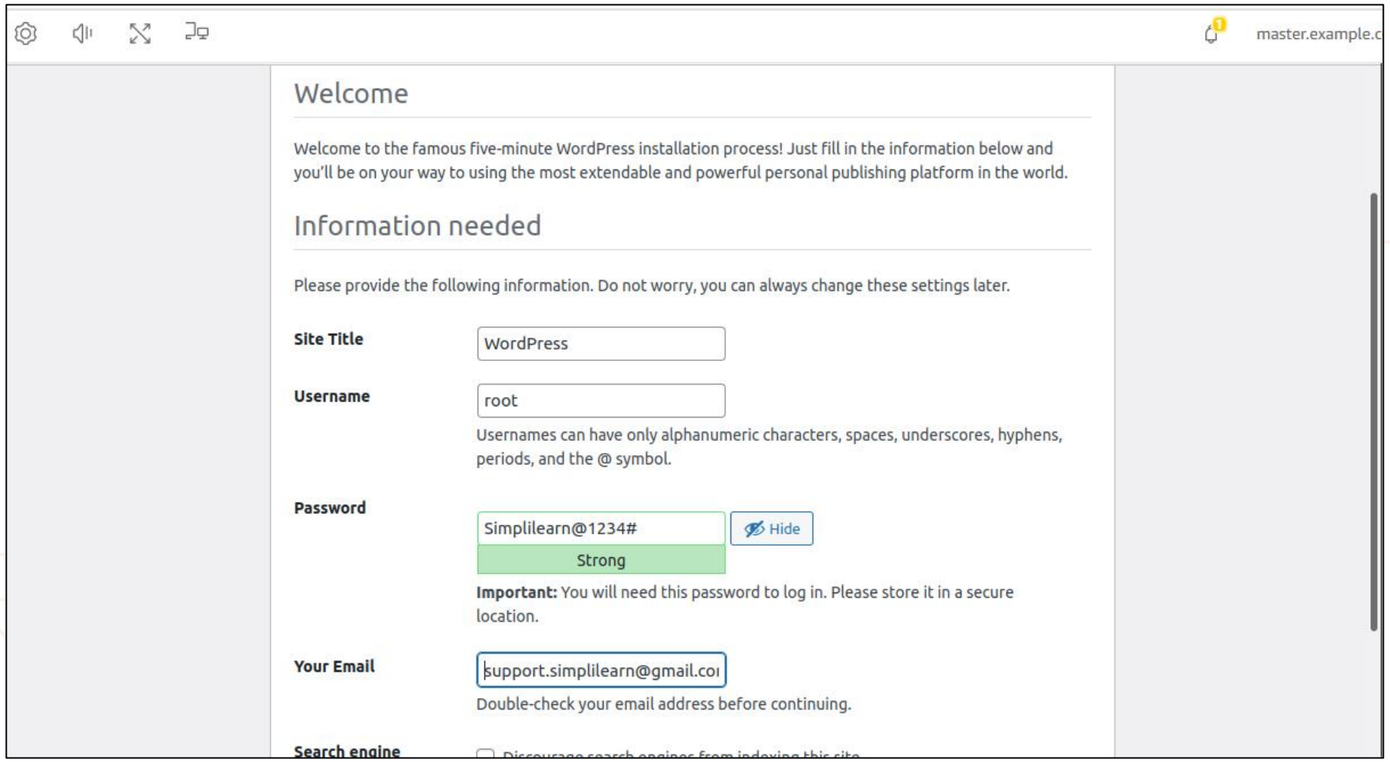
kubectl -n kubernetes-dashboard describe secret $(kubectl -n kubernetes-dashboard get secret | grep sandry | awk '{print $1}')

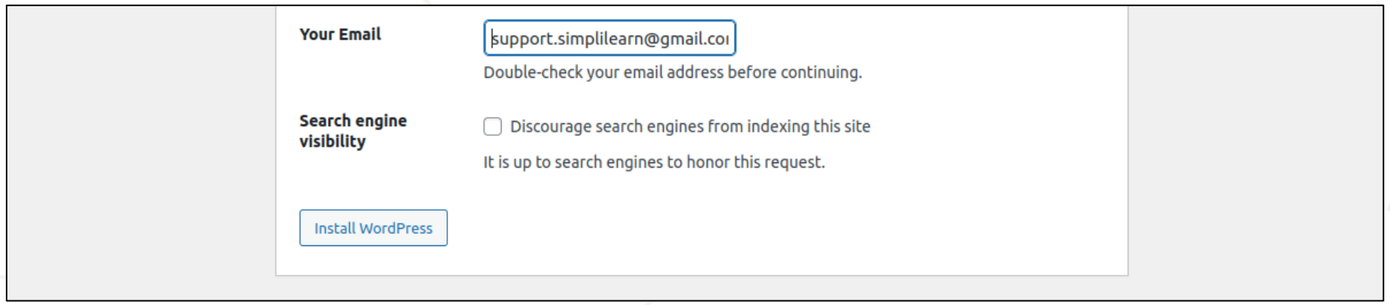
- **Task 4:** Configure the NFS-server for MySQL and WordPress Deployment, refer to the lesson-end project 7

Project Reference

- **Task 5:** To set up the NFS Client side for two applications separately, refer to the Lesson-end Project 7
- **Task 6:** To create and verify the PV, refer to the Lesson 7; demo 2
- **Task 7:** To create a secret for MySQL Deployments secret data, use the namespace cep-project1 in the Kubernetes dashboard
- **Task 8:** To create a configmap for WordPress Deployment to store non-sensitive information, use the namespace cep-project1 in the Kubernetes dashboard

Output Screenshot





Thank you