## Lab 2: Deploying applications with Helm and Amazon S3

Helm is a tool that helps manage Kubernetes applications. It simplifies the deployment and management of applications on Kubernetes by using Helm charts.

**Helm Chart**: A collection of YAML files that describe the resources needed to run an application in Kubernetes. It's like a recipe that tells Kubernetes what to deploy and how to configure it.

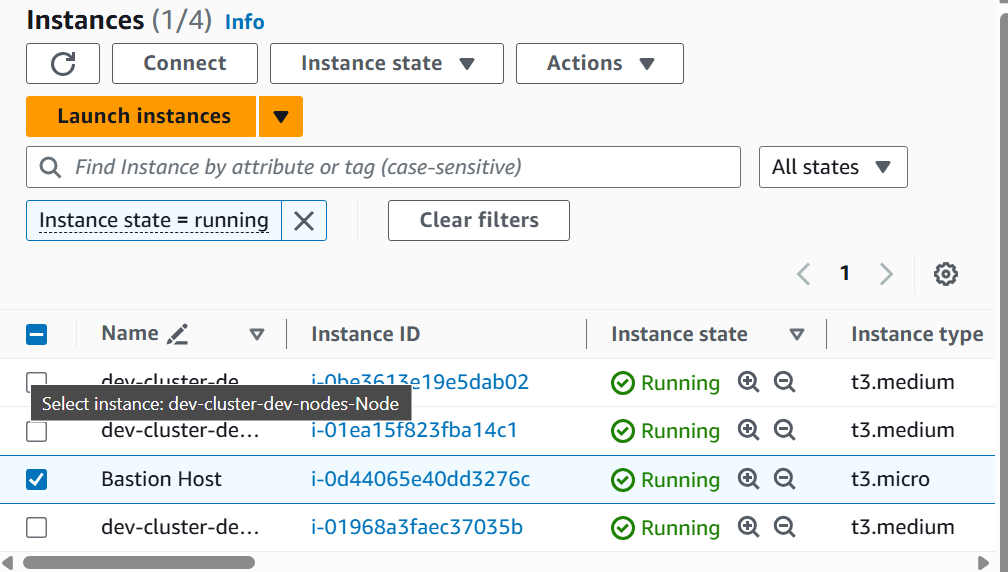
**Charts**: Contain templates that make the YAML files dynamic, allowing you to customise the configuration for different environments without changing the files themselves.

Objectives:

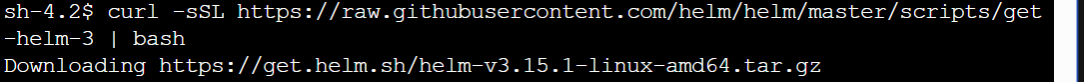
1. Define the basic structure of the Helm chart structure.
2. Set up an Amazon S3 bucket to use as a Helm repository.
3. Create and upload a chart to the S3 repository.
4. Package and install a Helm chart from an S3 repository.

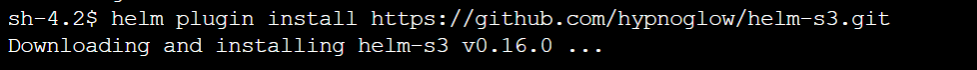
**Task 1: Install Helm and create an Amazon S3 bucket as a Helm repository**

1.1 Open EC2, go to running instances, and connect to Bastion Host

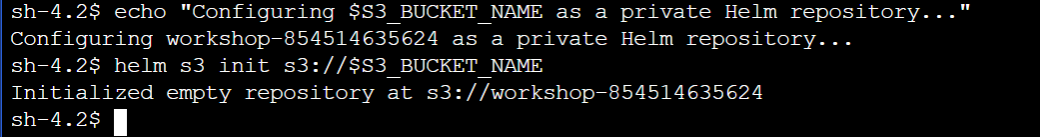
****

1.2 To install Helm, at the command prompt, enter the following command:

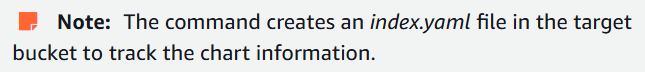




1.3 An Amazon S3 bucket for your hosted Helm repository has been pre-created by the lab and its name has been saved to the bastion host as an environment variable. To initialise the bucket as a Helm repository, enter the following command:



1.4 Verify the creation of index.yaml file  
index - This file keeps an index of all the charts in the repository.

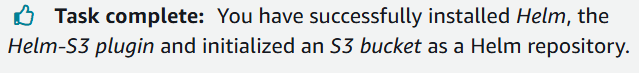




1.5 To add the S3 bucket as a chart repository for the Helm client, enter the following command:



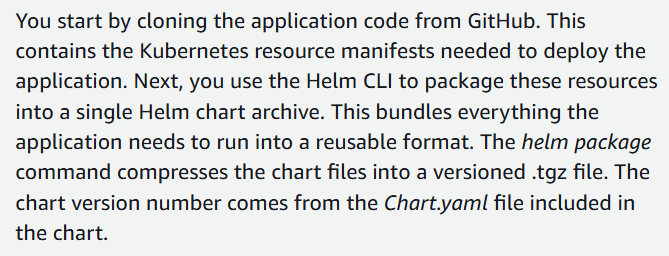
A Helm chart repository is a place where Helm charts are stored and from which they can be fetched and installed.



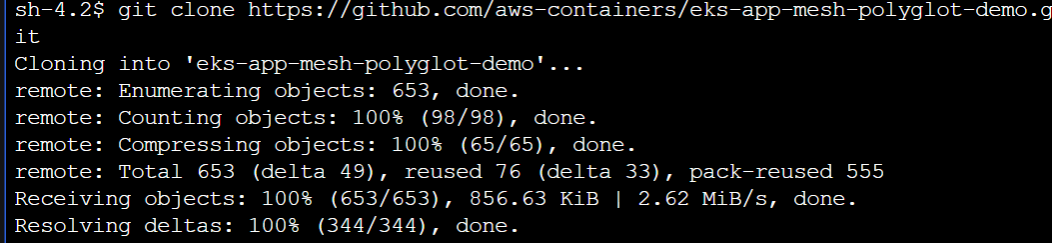
**Task 2: In this task, you package a sample application into a Helm chart and push it to the Amazon S3 repository that you created.**

**Doubt - Did we clone Helm Chart too, or did we create a Helm Chart from/for the Application Code.**

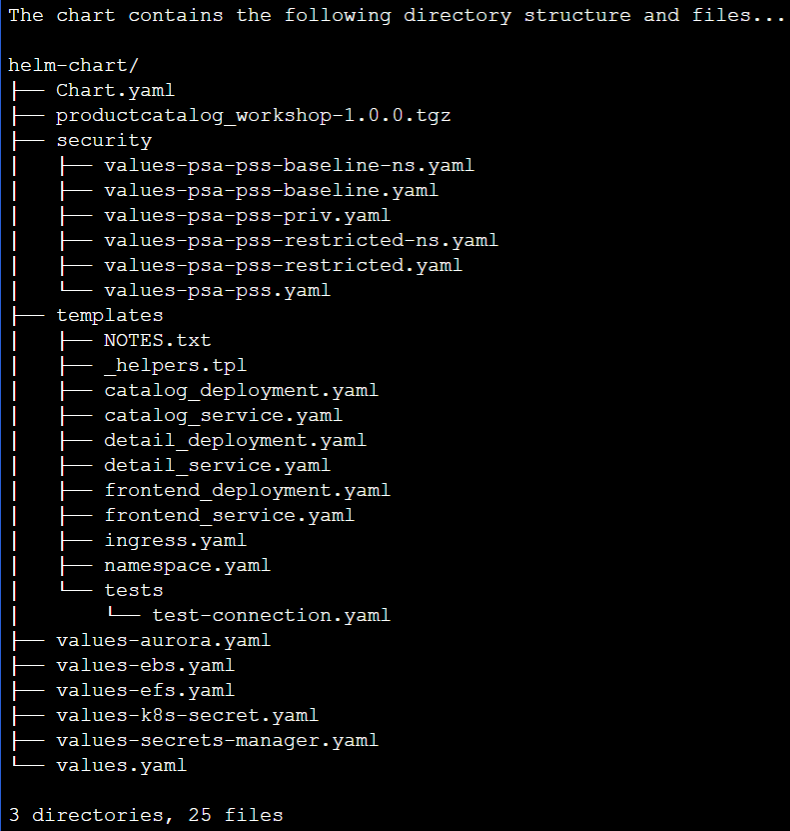
Overview:



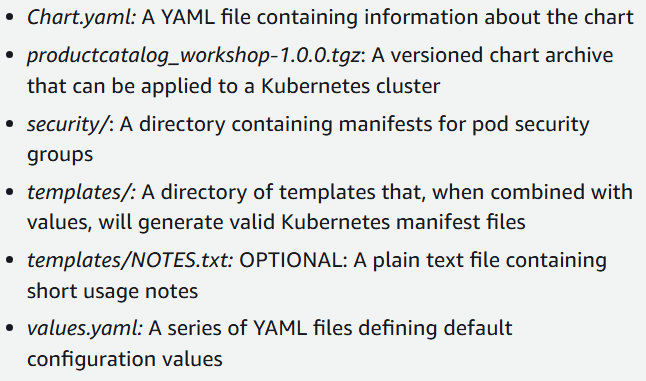
2.1 To clone the application Helm chart from the Containers on AWS Github page, enter the following command:



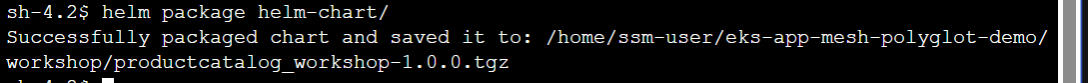
https://github.com/aws-containers/eks-app-mesh-polyglot-demo.git

2.2 The files in the helm chart directory:  


1. The directory name is the name of the chart



2.3 To package the application Helm chart, enter the following command:



Our Helm Chart has been packaged:

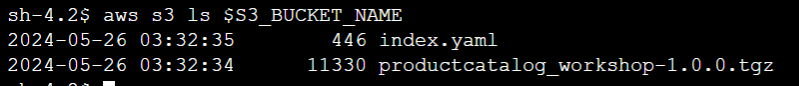


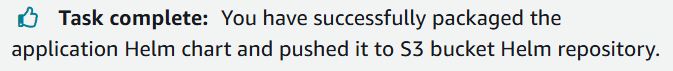
2.4 To push the packaged application Helm chart to the S3 Helm repo, enter the following command:



2.5 Verify the contents of the S3 bucket (helm repo), to verify that the application Helm chart was pushed to the S3 bucket successfully:

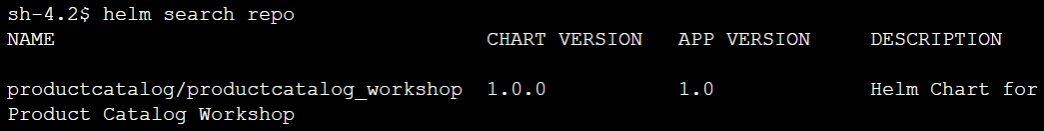
Using this command:





**Task 3: Search and install a Helm chart from the Amazon S3 repository**

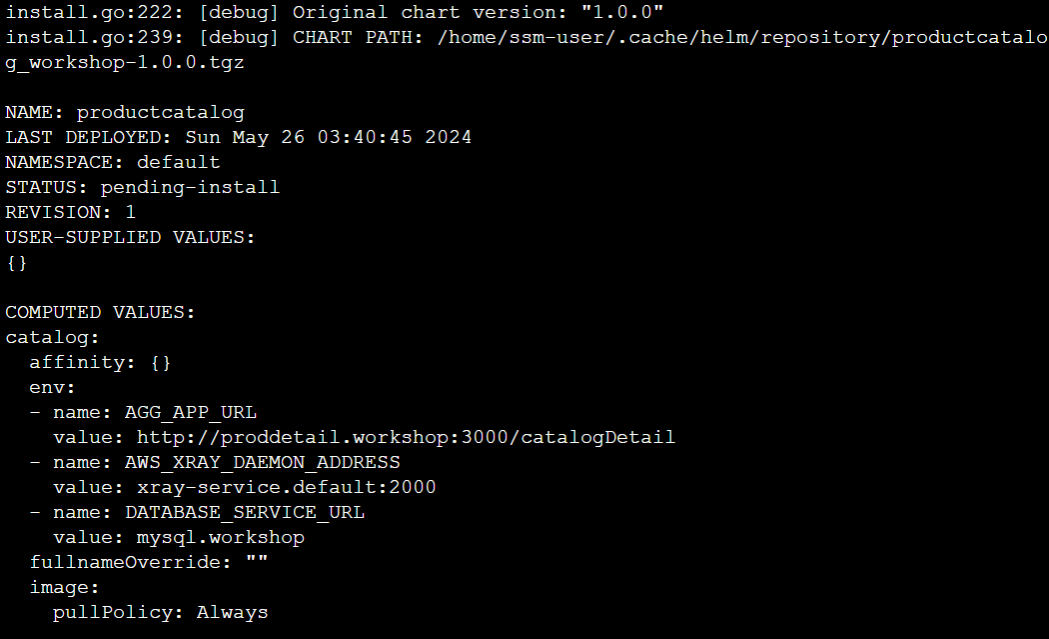
3.1 Use this to check the version of your helm chart



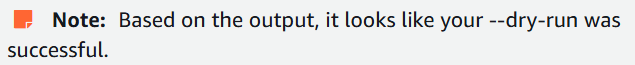
3.2 Use the “dry run” to test installing the chart without deploying any resources:



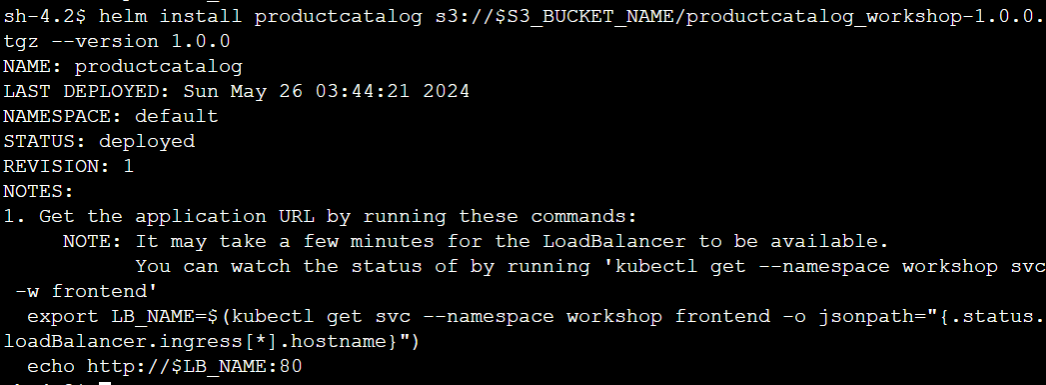
Truncated output:



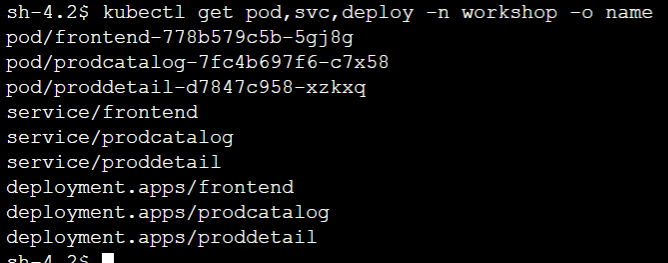
Note from the lab:



3.3 To install the Helm chart, enter the following command:

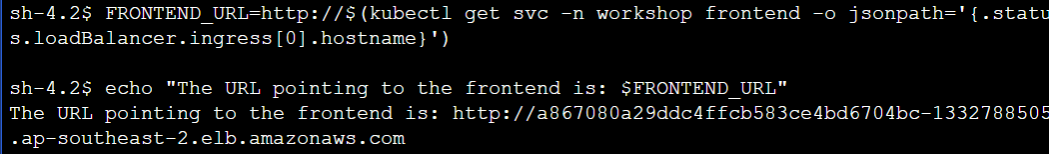


3.4 To list the resources deployed by the chart, enter the following command:



Helm chart has installed 3 pods, 3 services, and 3 deployments representing the frontend, productcatalog, and productdetail components of the application.

3.5 To view the frontend service:



Open the link

3.6 