1) What is Helm?

Helm Charts assist you in defining, installing, and upgrading even the most complicated Kubernetes application. … Helm is a CNCF-graduated project that is supported by the Helm community.

2) What are the advantages of Helm?

Helm helps in three ways: increases productivity, reduces the complexity of microservice deployments, and allows for the adoption of cloud-native applications.

3) Why to use Helm?

It can be time consuming and tedious to write and maintain Kubernetes YAML manifests for all of the essential Kubernetes objects. You'll need at least three YAML manifests with duplicated and hardcoded values for the most basic deployments. Helm streamlines the process by generating a single package that can be distributed across your cluster.

Helm is a client/server application that relied on Tiller (the helm server) to be installed in your cluster. Helm is similar to RPM and DEB packages in Linux, and it allows developers to package and ship applications to their end customers for installation.

Once you've installed and configured Helm (details below), you may use one simple helm install command to install production-ready apps from software providers like MongoDB, MySQL, and others into your Kubernetes cluster.

4) What are Helm Charts?

Helm Charts is used for combining all the Kubernetes YAML Manifests in a single package which can also be advertised to our Kubernetes Cluster. Once we install Helm CHart in our cluster it is easy as running one single helm, that simplifies the deployment of the containerized Applications.

5) What is the Folder Structure of Helm Chart?

Folder Structure of the Helm Charts are as follows:
Chart.yaml contains all the information about the charts.
LICENSE used in containing the License for the chart.
README.md is a readable file.
values.yaml used in configuring the values for the chart.
values.schema.json is a schema used in imposing the structure of values.yaml file.
charts/ used in containing charts which are depends on this chart.
crds/ is a custom resource definition.
templates/ are used in combining directory of templates with the values and in generating valid kubernetes manifest files.
templates/NOTES.txt used in containing Short Usage Notes.

6) Which of the different types of fields needed to be specified for dependencies in Helm Chart?

Different types of fields needed are as follows:
name
version
repository

7) How can we create Helm Charts?

We can create charts by using a specific structure.We need to use the command OUR-CHART-NAME for creating a Helm Chart.
OUR-CHART-NAME/

- .helmignore
- Chart.yaml
- values.yaml
- charts/
- templates/

8) What is Helm?

Helm is used as a package manager which helps in running atop Kubernetes and allowing the application structure through convenient Helm Chart, also managed with simple commands.

9) How does Helm work?

Helm is used in deploying charts as a package application, they are a collection of our pre configured application resources that can be deployed as a unit.We can deploy another version of its charts with different sets of configuration.

10) What are the concepts used in Helm?

Concepts used by Helm are:
Chart – It is a package consists of pre configured Kubernetes Resources.
Release – It is an instance that can be deployed to the Cluster with the help of Helm.
Repository – It is a group of charts that are available for others.

11) How can we install a specific Chart version in Helm?

We can use Prometheus in installing a specific chart version:

```
helm install -f source/prometheus/values.yaml prometheus –name source –namespace –version 6.7.4
```

12) How can we set multiple values with Helm?

We can set multiple values by using the following helm command:

```
helm install–set favoriteFood=junk./mychart
```

13) How does Helm updates Kubernetes?

Helm updates Kubernetes clusters by using the command given:

```
helm upgrades -d ingress-controller/values.yml nginx-ingress stable/nginx-ingress
```

14) How do we list all the available charts under a Helm Repo?

We can list all the charts by using the following command:

```
helm search repo [namespace]
```

15) How do we validate Helm Chart content?

We can validate Helm Chart contents by using the following command:

```
helm install –abcd –debug ./mychart
```

16) How can we uninstall Helm Chart on specific resource?

We can uninstall Helm Chart by using the following command, we should not use slash(/) in the command:
$ helm delete redis –example

17) Does helm still use Tiller?

With Tiller now gone, the security model for Helm is radically simplified. By removing Tiller, Helm 3 supports all the modern security, identity, and authorization features of modern Kubernetes.

18) What is replicaCount in helm?

Command-line parameter. Cluster management console field. compute.replicaCount. Replica count. Number of pod replicas for the compute container.

19) What is _helpers TPL in helm?

tpl? Helm allows for the use of Go templating in resource files for Kubernetes. A file named _helpers.tpl is usually used to define Go template helpers with this syntax: {{- define "yourFnName" -}} {{- printf "%s-%s" .Values.name .Values.version | trunc 63 -}} {{- end -}}

20) Can a helm chart have multiple deployments?

Take Your Helm Charts to the Next Level

Let's assume you're deploying a database with Kubernetes—including multiple deployments, containers, secrets, volumes, and services. Helm allows you to install the same database with a single command and a single set of values.

21) Where do you store helm charts?

All template files are stored in a chart's templates/ folder. When Helm renders the charts, it will pass every file in that directory through the template engine.

22) Where do you store helm charts?

All template files are stored in a chart's templates/ folder. When Helm renders the charts, it will pass every file in that directory through the template engine.

23) Where are helm repos stored locally?

The official Helm repo URL is https://kubernetes-charts.storage.googleapis.com . This repo is mainteined on GitHub and it’s URL is https://github.com/helm/charts. So the best approach is to clone the official repo github and work on it locally.

24) What is Helm values yaml?

yaml. All Helm packed applications have an associated values. yaml file which dictates the configuration of an application. By design many applications ship with a default values.

25) What is Helm Kubernetes?

What is Helm? In simple terms, Helm is a package manager for Kubernetes. Helm is the K8s equivalent of yum or apt. Helm deploys charts, which you can think of as a packaged application. It is a collection of all your versioned, pre-configured application resources which can be deployed as one unit.

26) What is Helm DevOps?

Helm is a package manager for Kubernetes that allows developers and operators to more easily package, configure, and deploy applications and services onto Kubernetes clusters. Helm is the best way to find, share, and use software built for Kubernetes.

27) Who invented helm?

Deis
Originally developed by Deis, Helm shortly became a de-facto open source standard for running and managing applications with Kubernetes. Imagine Kubernetes as an Operating System (OS), Helm is the apt or yum for it.

28) What are the benefits of Helm?

Benefits of Helm:
Helm Charts provide the ability to leverage Kubernetes packages through the click of a button or single CLI command. …
Helm charts are built atop Kubernetes. …
Another distinguishing feature Helm offers is the ability to customize application configurations during deployment.

29) Is helm an open source?

Helm is now an official Kubernetes project and is part of the Cloud Native Computing Foundation, a non-profit that supports open source projects in and around the Kubernetes ecosystem.

30) What is Helm deployment?

Helm is a Kubernetes deployment tool for automating creation, packaging, configuration, and deployment of applications and services to Kubernetes clusters. Kubernetes is a powerful container-orchestration system for application deployment.

31) What is Helm hub?

This hub provides a means for you to find charts hosted in many distributed repositories hosted by numerous people and organizations. Helm repositories can be hosted in many ways including as GitHub or GitLab pages, in object storage, using Chartmuseum, and via a service provider.

32) What is helm artifact?

Artifact Hub is a web-based application that enables finding, installing, and publishing packages and configurations for CNCF projects, including publicly available distributed charts Helm charts. It is a Cloud Native Computing Foundation sandbox project.

33) What is helm chart used for?

Helm Charts allow software vendors and developers to preconfigure their applications with sensible defaults. Charts also allow users/deployers to change parameters (e.g., resource limits for CPU and memory) of the application/chart using a consistent interface.

34) How do you push a helm chart?

To push a Helm chart to an Amazon ECR repository
Install the latest version of the Helm client. …
Currently, OCI support is considered experimental. …
Use the following steps to create a test Helm chart. …
Package the chart. …
Create a repository to store your Helm chart.

35) What is Umbrella chart in helm?

Helm charts have the ability to include other charts, referred to as subcharts, via their dependencies section. When a chart is created for the purpose of grouping together related subcharts/services, such as to compose a whole application or deployment, we call this an umbrella chart.

36) How does helm upgrade work?

When a new version of a chart is released, or when you want to change the configuration of your release, you can use the helm upgrade command. An upgrade takes an existing release and upgrades it according to the information you provide.

37) Does helm have API?

Helm API. Helmit provides a Go API for managing Helm charts within a Kubernetes cluster. Tests, benchmarks, and simulations can use the Helmit Helm API to configure and install charts to test and query resources within releases.

38) What is Helm init?

Synopsis. The Kubernetes package manager. To begin working with Helm, run the ‘helm init’ command: $ helm init. This will install Tiller to your running Kubernetes cluster. It will also set up any necessary local configuration.

39) What is value yaml?

The values. yaml file is used to pass values into the Release helm chart. The file contains default parameters that you can override. The values. yaml file is typically located in the folder where you extracted the Release Helm chart zip file.

40) What are annotations in Kubernetes?

Annotations allow you to add non-identifying metadata to Kubernetes objects. Examples include phone numbers of persons responsible for the object or tool information for debugging purposes. In short, annotations can hold any kind of information that is useful and can provide context to DevOps teams.

41) What is fullname Helm?

fullnameOverride completely replaces the generated name. These come from the template provided by Helm for new charts. A typical object in the templates is named. name: {{ include “. fullname” . }}

42) What is Helm CRD?

When working with Custom Resource Definitions (CRDs), it is important to distinguish two different pieces: There is a declaration of a CRD. This is the YAML file that has the kind CustomResourceDefinition. Then there are resources that use the CRD.

43) What is Helm dependency update?

Update the on-disk dependencies to mirror Chart. yaml. This command verifies that the required charts, as expressed in ‘Chart. yaml’, are present in ‘charts/’ and are at an acceptable version. It will pull down the latest charts that satisfy the dependencies, and clean up old dependencies.

44) What does helm repo update do?

Update gets the latest information about charts from the respective chart repositories. Information is cached locally, where it is used by commands like ‘helm search’. You can optionally specify a list of repositories you want to update

45) What happened helm init?

The helm init command has been removed. It performed two primary functions. First, it installed Tiller. This is no longer needed.

46) What is Helm value?

In the previous section we looked at the built-in objects that Helm templates offer. One of the built-in objects is Values . This object provides access to values passed into the chart. Individual parameters passed with –set (such as helm install –set foo=bar ./mychart)

47) Where are helm values stored?

Where this information is stored depends on the version of helm you are using: For version 2: it is in a configMap named . , in the kube-system namespace. You can get more detail on that here.

48) Who created helm Kubernetes?

DeisLabs
Helm is an open-source project which was originally created by DeisLabs and donated to CNCF, which now maintains it. The original goal of Helm was to provide users with a better way to manage all the Kubernetes YAML files we create on Kubernetes projects.

49) What is helm Azure?

Helm is an open-source packaging tool that helps you install and manage the lifecycle of Kubernetes applications. Similar to Linux package managers such as APT and Yum, Helm is used to manage Kubernetes charts, which are packages of preconfigured Kubernetes resources.

50) What is helm package manager?

Helm is an application package manager for Kubernetes that you use to standardize and simplify the deployment of cloud-native applications on Kubernetes. Here you’ll see how to install thirdparty packages called Helm charts and how to create and install Helm charts for the workloads your teams develop.

51) What is Helm client and server?

The Helm Client is a command-line client for end users. The client is responsible for the following domains: Local chart development. Managing repositories. Interacting with the Tiller server.

52) What is helm github?

Helm is a tool for managing Charts. Charts are packages of pre-configured Kubernetes resources. Use Helm to: Find and use popular software packaged as Helm Charts to run in Kubernetes. Share your own applications as Helm Charts.