

=====

K8S HELM

=====

-> We deployed our apps in Kubernetes cluster using Manifest files

-> Manifest files we can write in 2 ways

1) JSON

2) YML (more demand)

-> It is difficult to write manifest files for our applications

-> Helm is a package manager for k8s applications

-> Helm allows you to install or deploy applications on kubernetes cluster in a similar manner to yum/apt for linux distributions.

-> Helm lets you fetch, deploy and manage the lifecycle of applications both 3rd party apps and your own applications

Ex: prometheus, grafana, nginx-ingress, ELK stack are third party apps

-> Helm introduces several familiar concepts such as

Helm Chart (package contains k8s manifests - templates)

Helm Repositories which holds helm charts/packages

A CLI with install/upgrade/remove commands

+++++

Why to use Helm?

+++++

-> Deploying application on K8S cluster is little difficult

-> As part of app deployment we need to create below k8s objects

Deployment

Service

ConfigMaps/Secrets

Volumes

Ingress Rules

HPA

-> Helm greatly simplifies the process of creating, deploying and managing applications on k8s cluster

-> Helm also maintains a versioned history of every chart (application) installation. If something goes wrong, you can simply call 'helm rollback'.

-> Setting up a single application can involve creating multiple independent k8s resources and each resource requires a manifest file.

#####

What is Helm Chart

#####

-> HELM chart is basically just a collection of manifest files organized in a specific directory structure that describe a related K8S resource.

-> There are two main components in HELM chart

- 1) template
- 2) value

-> Templates and values renders a manifest which can understand by k8s

-> Helm uses charts to pack all the required k8s components (manifests) for an application to deploy, run and scale.

-> charts are very similar to RPM and DEB packages for Linux.

Ex: yum install git

Note: it will interact with repo and it will download git

```
#####
HELM Concepts
#####
```

-> Helm packages are called charts, and they consist of a few YML configuration files and some templates that are rendered into K8S manifest files. Here is the basic directory structure of a chart.

charts : dependent charts will be added here

templates: contains all template files

values : It contains values which are required for templates

```
#####
HELM Architecture
#####
```

```
what-the-helm
â”œâ”€â”€â”€ Chart.yaml
â”œâ”€â”€â”€ charts
â”œâ”€â”€â”€ templates
â””,   â”œâ”€â”€â”€â”€ NOTES.txt
â””,   â”œâ”€â”€â”€â”€ _helpers.tpl
â””,   â”œâ”€â”€â”€â”€ deployment.yaml
â””,   â”œâ”€â”€â”€â”€ ingress.yaml
â””,   â”œâ”€â”€â”€â”€ service.yaml
â””,   â””â””â””â”€â”€ tests
â””,   â””â””â””â”€â”€ test-connection.yaml
â””â””â””â”€â”€ values.yaml
```

```
#####
Helm Installation
#####
```

```
$ curl -fsSl -o get_helm.sh https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3
```

```
$ chmod 700 get_helm.sh
```

```
$ ./get_helm.sh
```

```
$ helm
```

-> check do we have metrics server on the cluster

```
$ kubectl top pods
```

```
$ kubectl top nodes
```

```
# check helm repos
```

```
$ helm repo ls
```

```
# Before you can install the chart you will need to add the metrics-server repo to helm
```

```
$ helm repo add metrics-server https://kubernetes-sigs.github.io/metrics-server/
```

```
# Install the chart
```

```
$ helm upgrade --install metrics-server metrics-server/metrics-server
```

```
$ helm list
```

```
$ helm delete <release-name>
```

```
=====
```

```
Metric Server Unavailability issue fix
```

```
=====
```

```
https://www.linuxsysadmins.com/service-unavailable-kubernetes-metrics/
```