CKA-helm

讲师： 老段 RHCE/RHCA/COA/CKA

了解heml

helm的作用就是把许多的定义比如svc，比如deployment，比如securt一次性全部定义好，放在源里统一管理

这样很容器在其他机器上部署

安装helm

#curl https://raw.githubusercontent.com/kubernetes/helm/master/scripts/get | bash

此语句没法正常下载安装helm

提前下载所需要的文件
https://kubernetes-helm.storage.googleapis.com/helm-v2.11.0-linux-amd64.tar.gz
https://kubernetes-helm.storage.googleapis.com/helm-v2.11.0-linux-amd64.tar.gz.sha256

wget https://raw.githubusercontent.com/kubernetes/helm/master/scripts/get

修改get脚本

安装helm

```
112 downloadFile() {
113   HELM_DIST="helm-$TAG-$OS-$ARCH.tar.gz"
114   DOWNLOAD_URL="https://kubernetes-helm.storage.googleapis.com/$HELM_DIST"
115   CHECKSUM_URL="$DOWNLOAD_URL.sha256"
116   HELM_TMP_ROOT="$(mktemp -dt helm-installer-XXXXXX)"
117   HELM_TMP_FILE="$HELM_TMP_ROOT/$HELM_DIST"
118   HELM_SUM_FILE="$HELM_TMP_ROOT/$HELM_DIST.sha256"
119   echo "Downloading $DOWNLOAD_URL"
120   cp helm* $HELM_TMP_ROOT
121 #  if type "curl" > /dev/null; then
122 #    curl -SsL "$CHECKSUM_URL" -o "$HELM_SUM_FILE"
123 #  elif type "wget" > /dev/null; then
124 #   wget -q -O "$HELM_SUM_FILE" "$CHECKSUM_URL"
125 #  fi
126 #  if type "curl" > /dev/null; then
127 #   curl -SsL "$DOWNLOAD_URL" -o "$HELM_TMP_FILE"
128 #  elif type "wget" > /dev/null; then
129 #   wget -q -O "$HELM_TMP_FILE" "$DOWNLOAD_URL"
130 #  fi
131 }
```

在87行定义TAG=v2.11.0

bash get

查看版本
helm version

helm completion bash > ~/.hemlrc; echo "source ~/.hemlrc" >> ~/.bashrc

安装tiller

#helm init --service-account tiller --tiller-image registry.us-east-1.aliyuncs.com/acs/tiller:v2.11.0

#helm init --upgrade

kubectl create serviceaccount --namespace kube-system tiller

kubectl create clusterrolebinding tiller-cluster-rule --clusterrole=cluster-admin --serviceaccount=kube-system:tiller

helm init --service-account tiller --tiller-image registry.us-east-1.aliyuncs.com/acs/tiller:v2.11.0 --stable-repo-url https://kubernetes.oss-cn-hangzhou.aliyuncs.com/charts

helm的基本操作

查看源
helm repo list

helm search

helm search redis

helm search mysql

实例讲解

helm install stable/mysql

直接这样安装的话，所有的一切都是默认值，包括镜像，pvc等，对用户来说并不可控

查看所有的对象 kubectl get all
删除所有

讲解chart
上述命令执行之后会在/root/.helm/cache/archive有一个压缩包

Chart.yaml 描述了chart的概要
values.yaml 定义了所需的各种参数，包括镜像、pv等等用户可以自定义的参数

使用helm安装应用程序

修改values.yaml

helm install .

或者

helm inspect values stable/mysql > z.yaml

helm install --values=z.yaml stable/mysql

自定义chart

helm create mychart

检测语法

helm lint mychart

helm install --dry-run mychart --debug

helm install mychart

推送到源里

打包
helm package mychart

mkdir myrepo

mv mychart-0.1.0.tgz myrepo

docker run -dit --name=c1 -p 8080:80 -v /data:/usr/share/nginx/html docker.io/nginx

helm repo index myrepo/ --url http://192.168.26.52:32419/charts

然后把myrepo拷贝过去 cp myrepo/* /data/charts/

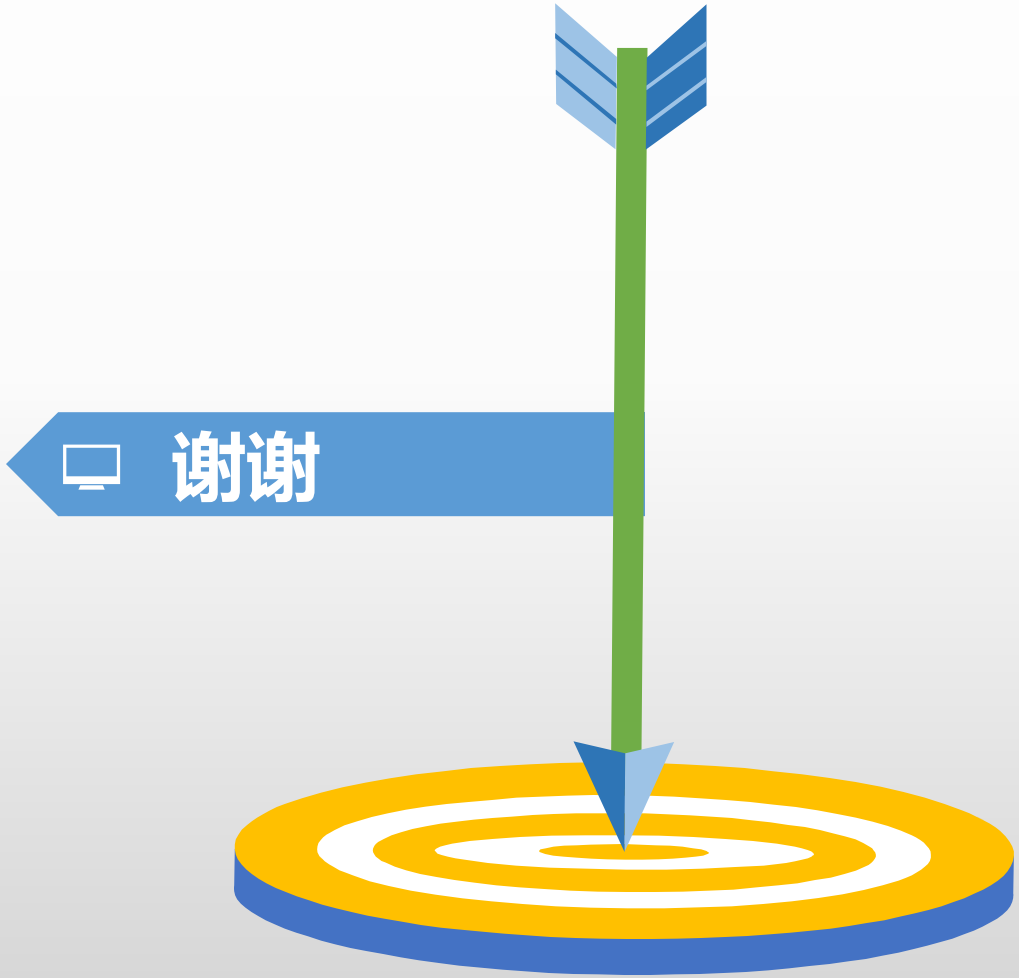
用私有源里部署应用程序

helm repo add myrepo http://192.168.26.52:8080/charts

helm repo remove myrepo

helm search mychart

helm install myrepo/mychart
helm install myrepo/mysql


谢谢