链滴

kubernetes1.13.1 安装 helm 详细教程

作者：18582596683

原文链接：https://ld246.com/article/1547820454417

来源网站：链滴

1.安装准备

(1).默认k8s已正常运行
(2).准备安装helm的机器已配置好kubectl和kubeconfig文件，确保kubectl工具可以在这台机器上访apiserver且正常使用。
(3).这里我在k8s集群的master上安装helm，已配置好环境。

2.下载helm命令行工具到master节点k8s01

wget https://storage.googleapis.com/kubernetes-helm/helm-v2.12.2-linux-amd64.tar.gz

解压并复制到/usr/local/bin/
tar -zxvf helm-v2.12.2-linux-amd64.tar.gz
cd linux-amd64/
cp helm /usr/local/bin/

3.创建配置文件

###说明：
因为Kubernetes APIServer开启了RBAC访问控制，所以需要创建tiller使用的service account: tille并分配合适的角色给它。 详细内容可以查看helm文档中的[Role-based Access Control](https://doc.helm.sh/using_helm/#role-based-access-control)。 这里简单起见直接分配cluster-admin这个群内置的ClusterRole给它。创建rbac-config.yaml文件：

```
[root@k8s01 helm-amd64]# cat rbac-config.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: tiller
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRoleBinding
metadata:
  name: tiller
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
  - kind: ServiceAccount
    name: tiller
    namespace: kube-system
```

4.执行配置文件，创建ServiceAccount

```
[root@k8s01 helm-amd64]# kubectl create -f rbac-config.yaml
serviceaccount/tiller created
```

clusterrolebinding.rbac.authorization.k8s.io/tiller created

5.部署tiller

```
##由于不能访问到google，这里使用阿里云的tiller镜像
[root@k8s01 helm-amd64]# docker pull registry.cn-hangzhou.aliyuncs.com/google_container
/tiller:v2.12.2
v2.12.2: Pulling from google_containers/tiller
407ea412d82c: Pull complete
660131a755cd: Pull complete
84da816f419d: Pull complete
2572e90dc0d8: Pull complete
Digest: sha256:547fe38057d538b26236a6432b8e00ce52f627a5da11b187a8b601e387afdfa2
Status: Downloaded newer image for registry.cn-hangzhou.aliyuncs.com/google_containers/til
er:v2.12.2

[root@k8s01 helm-amd64]# helm init --service-account tiller --skip-refresh --tiller-image=reg
istry.cn-hangzhou.aliyuncs.com/google_containers/tiller:v2.12.2
Creating /root/.helm
Creating /root/.helm/repository
Creating /root/.helm/repository/cache
Creating /root/.helm/repository/local
Creating /root/.helm/plugins
Creating /root/.helm/starters
Creating /root/.helm/cache/archive
Creating /root/.helm/repository/repositories.yaml
Adding stable repo with URL: https://kubernetes-charts.storage.googleapis.com
Adding local repo with URL: http://127.0.0.1:8879/charts
$HELM_HOME has been configured at /root/.helm.

Tiller (the Helm server-side component) has been installed into your Kubernetes Cluster.

Please note: by default, Tiller is deployed with an insecure 'allow unauthenticated users' policy

To prevent this, run `helm init` with the --tiller-tls-verify flag.
For more information on securing your installation see: https://docs.helm.sh/using_helm/#se
uring-your-helm-installation
Happy Helming!
```

6.查看helm状态

```
[root@k8s01 helm-amd64]# helm version
Client: &version.Version{SemVer:"v2.12.2", GitCommit:"7d2b0c73d734f6586ed222a567c5d103
ed435be", GitTreeState:"clean"}
Server: &version.Version{SemVer:"v2.12.2", GitCommit:"7d2b0c73d734f6586ed222a567c5d10
fed435be", GitTreeState:"clean"}
[root@k8s01 helm-amd64]# kubectl get pod -n kube-system -l app=helm
NAME                            READY   STATUS    RESTARTS   AGE
tiller-deploy-7d6b75487c-fz7j9   1/1     Running   0          3m47s
[root@k8s01 helm-amd64]# helm version
Client: &version.Version{SemVer:"v2.12.2", GitCommit:"7d2b0c73d734f6586ed222a567c5d103
```

ed435be", GitTreeState:"clean"}
Server: &version.Version{SemVer:"v2.12.2", GitCommit:"7d2b0c73d734f6586ed222a567c5d10fed435be", GitTreeState:"clean"}

OK,helm安装成功！！！