# Helm

The Kubernetes Package Manager

# Installing Helm

There are two parts to Helm: The Helm client (`helm`) and the Helm server (`Tiller`).

## Installing the Helm Client

The Helm client can be installed either from source, or from pre-built binary releases.

### From Snap (Linux)

The Snap package for Helm is maintained by Snapcrafters.

```
sudo snap install helm --classic
```

```
helm 2.12.0 from Snapcrafters installed
```

# Initialize Helm and Install Tiller

## Role-based Access Control

In Kubernetes, granting a role to an application-specific service account is a best practice to ensure that your application is operating in the scope that you have specified.

### Tiller and Role-Based Access Control

You can add a service account to Tiller using the `--service-account <NAME>` flag while you're configuring Helm. As a prerequisite, you'll have to create a role binding which specifies a role and a service account name that have been set up in advance.

Once you have satisfied the pre-requisite and have a service account with the correct permissions, you'll run a command like this: `helm init --service-account <NAME>`

**Example: Service account with cluster-admin role**

First create a service account and attach `cluster-admin` role to it. This enables the tiler pod to communicate with the kubernetes api

```
cat << EOF > rbac-config.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
  name: tiller
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: tiller
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
  - kind: ServiceAccount
    name: tiller
    namespace: kube-system
EOF
```

> ***Note*** *The cluster-admin role is created by default in a Kubernetes cluster, so you don't have to define it explicitly.*

```
kubectl create -f rbac-config.yaml
```

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

```
helm init --service-account tiller
```

This will install Tiller into the Kubernetes cluster with `kubectl config current-context`.

```
Creating /home/admatic/.helm
Creating /home/admatic/.helm/repository
Creating /home/admatic/.helm/repository/cache
Creating /home/admatic/.helm/repository/local
Creating /home/admatic/.helm/plugins
Creating /home/admatic/.helm/starters
Creating /home/admatic/.helm/cache/archive
Creating /home/admatic/.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 /home/admatic/.helm.
```

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

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_h
elm/#securing-your-helm-installation
Happy Helming!
```

# Verify helm

```
kubectl get deploy,svc tiller-deploy -n kube-system
```

```
NAME                                  DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   A
GE
deployment.extensions/tiller-deploy   1         1         1            0           3
s

NAME                    TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)     AGE
service/tiller-deploy   ClusterIP   10.7.247.199   <none>        44134/TCP   3s
```