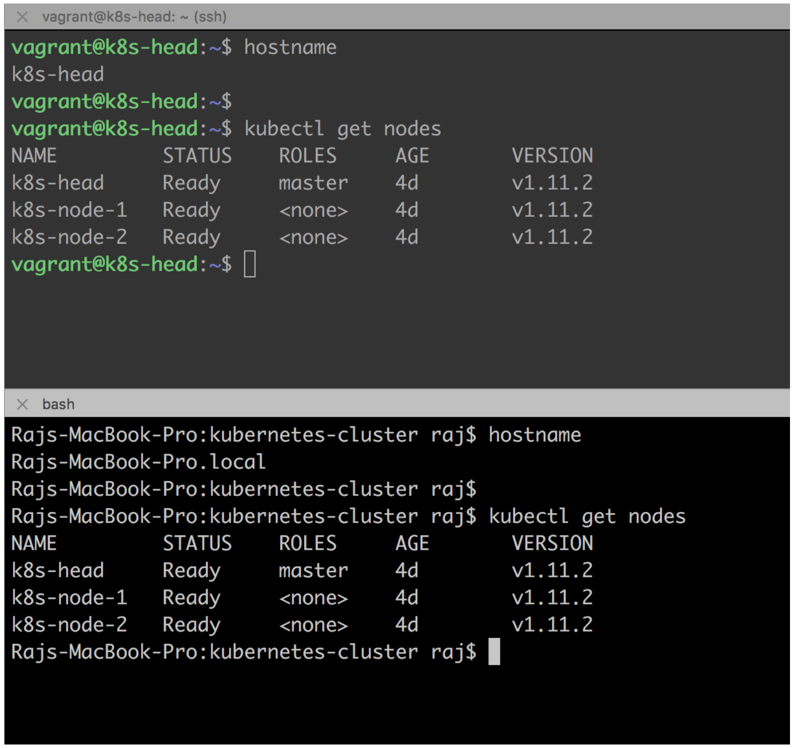
Configure Local Kubectl to Access Remote Kubernetes Cluster

Raj Rajaratnam Follow

Sep 9, 2018 · 2 min read

In my previous post, I explained how to configure a multi node Kubernetes cluster using Vagrant, VirtualBox and Kubeadm. Today I'm trying to get my local Kubectl running on my Macbook to access the remote Kubernetes cluster running on VirtualBox. This way I don't need to ssh into Kubernetes master to run Kubectl. Very convenient. Here is how I did it.



```
vagrant@k8s-head:~$ hostname
k8s-head
vagrant@k8s-head:~$
vagrant@k8s-head:~$ kubectl get nodes
NAME         STATUS    ROLES     AGE       VERSION
k8s-head     Ready     master    4d        v1.11.2
k8s-node-1   Ready     <none>    4d        v1.11.2
k8s-node-2   Ready     <none>    4d        v1.11.2
vagrant@k8s-head:~$
```

```
Rajs-MacBook-Pro:kubernetes-cluster raj$ hostname
Rajs-MacBook-Pro.local
Rajs-MacBook-Pro:kubernetes-cluster raj$
Rajs-MacBook-Pro:kubernetes-cluster raj$ kubectl get nodes
NAME         STATUS    ROLES     AGE       VERSION
k8s-head     Ready     master    4d        v1.11.2
k8s-node-1   Ready     <none>    4d        v1.11.2
k8s-node-2   Ready     <none>    4d        v1.11.2
Rajs-MacBook-Pro:kubernetes-cluster raj$
```

. . .

Step-1 : Install Kubectl in Macbook

You can install Kubectl in Macbook using Homebrew with the following command.

```
brew install kubernetes-cli
```

Now if you run `kubectl version` command in your Macbook terminal, you will get some errors like below.

```
Rajs-MacBook-Pro:.kube raj$ kubectl version
Client Version: version.Info{Major:"1", Minor:"11",
GitVersion:"v1.11.2",
GitCommit:"bb9ffb1654d4a729bb4cec18ff088eacc153c239",
GitTreeState:"clean", BuildDate:"2018-08-08T16:31:10Z",
GoVersion:"go1.10.3", Compiler:"gc",
Platform:"darwin/amd64"}
The connection to the server localhost:8080 was refused -
did you specify the right host or port?
```

Kubectl in my Macbook is trying to connect to the local Kubernetes master and failing because I do not have any local Kubernetes cluster running on my Macbook. Now let's see how I can point my Kubectl to a remote Kubernetes master.

. . .

Step-2 : Download Kubernetes Credentials From Remote Cluster

You need to first copy some Kubernetes credentials from remote Kubernetes master to your Macbook.

```
scp -r vagrant@192.168.205.10:/home/vagrant/.kube .
```

Step-3 : Copy Kubernetes Credentials To Your Home

Copy the Kubernetes Credentials your downloaded to your home directory as shown below.

```
cp -r .kube $HOME/
```

That’s all you have to do. Your local Kubectl should be able to connect with the remote Kubernetes cluster now.

· · ·

Now if you run `kubectl version` command in your Macbook terminal, you will get successfully connected to remote Kubernetes master and the output will be like this.

```
Rajs-MacBook-Pro:.kube raj$ kubectl version
Client Version: version.Info{Major:"1", Minor:"11",
GitVersion:"v1.11.2",
GitCommit:"bb9ffb1654d4a729bb4cec18ff088eacc153c239",
GitTreeState:"clean", BuildDate:"2018-08-08T16:31:10Z",
GoVersion:"go1.10.3", Compiler:"gc",
Platform:"darwin/amd64"}
Server Version: version.Info{Major:"1", Minor:"11",
GitVersion:"v1.11.2",
GitCommit:"bb9ffb1654d4a729bb4cec18ff088eacc153c239",
GitTreeState:"clean", BuildDate:"2018-08-07T23:08:19Z",
GoVersion:"go1.10.3", Compiler:"gc", Platform:"linux/amd64"}
```

· · ·

Run `kubectl get nodes` command in your Macbook terminal and verify you can list your remote cluster nodes.

```
Rajs-MacBook-Pro:kubernetes-cluster raj$ kubectl get nodes
NAME         STATUS    ROLES     AGE       VERSION
k8s-head     Ready     master    4d        v1.11.2
k8s-node-1   Ready     <none>    4d        v1.11.2
k8s-node-2   Ready     <none>    4d        v1.11.2
```

🙏