**[Steps for IoT Edge Virtual Kubelet Provider](https://github.com/azure/iot-edge-virtual-kubelet-provider)**

1. [Clone IoT Edge Virtual Kubelet Provider](#One)
2. [Create an IoT Hub](#Two)
3. [Deploy a Cluster (IE An AKS Cluster)](#Three)
4. [Create Secrets](#Four)
5. [Start IoT Edge Virtual Kubelet Provider Deployment](#Five)
6. [Test Pods and Nodes](#Six)
7. [Follow-up](#Seven)
8. **Login into portal.azure.com**

Ctrl+Click [Here](https://portal.azure.com/)

1. **Clone IoT Edge Virtual Kubelet Provider**
2. Open the Cloud Shell >\_ and select Bash
3. **Copy and Paste the Following:**

git clone https://github.com/Azure/iot-edge-virtual-kubelet-provider.git

1. **Create an IoT Hub**
2. In the search bar on the top, type “hub” and select IoT Hub
3. Click **+**Add
4. Create the Hub (Create new resource group, or use an existing one)
5. **Deploy a Kubernetes Cluster (IE: AKS)**

Follow the steps [Here](https://docs.microsoft.com/en-us/azure/aks/kubernetes-walkthrough) up to (and including) connecting to the cluster

Or… **Copy and Paste the Following:**

az aks create --resource-group <Resource Group> --name <New Cluster Name> --node-count 1 --generate-ssh-keys

az aks get-credentials --resource-group <Resource Group> --name <Cluster Name>

kubectl get nodes

1. **Create Kubernetes Secrets Store (To hold the IoT Hub Connection String)**
   1. Navigate to the IoT Hub resource in the Azure Portal
      1. Click Resource Groups
      2. Click on your newly created resource group (from the previous step)
      3. Click on the Hub Name
   2. In the scroll bar on the left, underneath the settings tab, click on Shared Access Policies
   3. Click on the tab that says iothubowner, and a window should pop up on the right side
   4. Copy and Paste the following into the shell, but do **NOT** yet hit enter:

kubectl create secret generic my-secrets \

--from-literal=hub0-cs='<iot-hub-owner-connection-string>'

* 1. Copy the ‘Connection String—Primary Key’ (or click the blue button to the right of it)
  2. Paste (shift-insert or right click) the string in replacement of <iot-hub-owner-connection-string> (replacing the < > symbols as well) and now hit enter

1. **Start IoT Edge Virtual Kubelet Provider Deployment**
   1. **Copy and Paste the Following:**

kubectl --namespace kube-system create serviceaccount tiller

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

helm init --service-account tiller

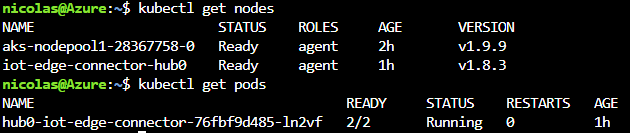
cd iot-edge-virtual-kubelet-provider/src/charts/iot-edge-connector/

helm install -n hub0 --set rbac.install=true .

Notes:

* Here, the rbac.install=true is only necessary if giving RBAC Authorization
* You may need to run: helm install -n hub0 src/charts/iot-edge-connector

1. **Test Nodes and Pods**

A successful deployment should look somewhat like this:

1. **Follow up**

There is an example deployment provided to test, which can be obtained by entering the following:

kubectl apply -f \

iot-edge-virtual-kubelet-provider/src/Microsoft.Azure.VirtualKubelet.Edge.Provider/sample-deployment.yaml

This is a sample thermometer. If successful, you will be able to find a deployment under the “IoT Edge Deployments” category under the Hub’s IoT Edge Section

1. **Clean up**

To delete everything you just did, we just have to delete the resource group. Everything attached to it with be removed

az group delete --name <resource-name> --yes --no-wait

Replacing the <resource-name> with the resource name you chose.

1. **Scaling**

To scale the number of nodes (IoT Edge, slow)

az aks scale -g myR -n myAKS -c <number-of-nodes>

To scale the number of pods (Kubernetes, fast)

Kubectl scale deployment/<deployment name> --replicas <number-of-pods>