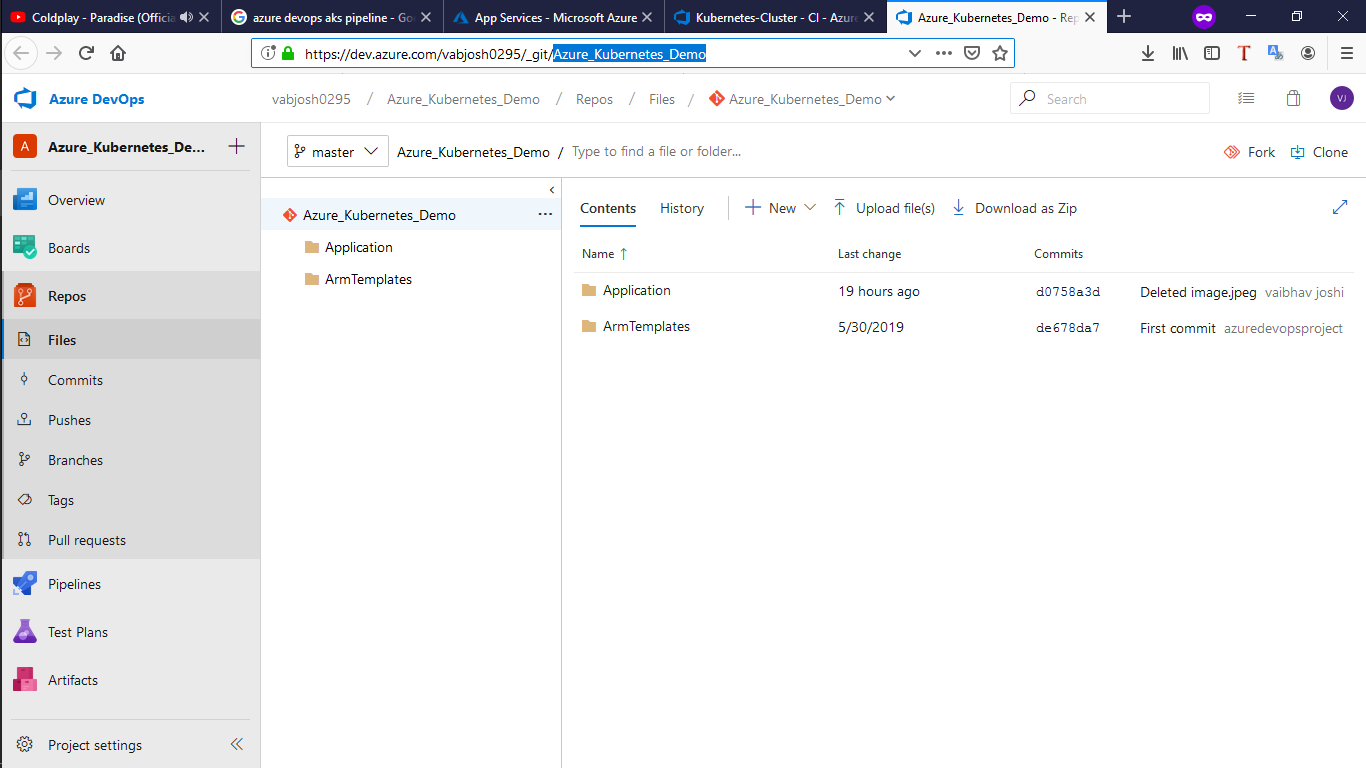
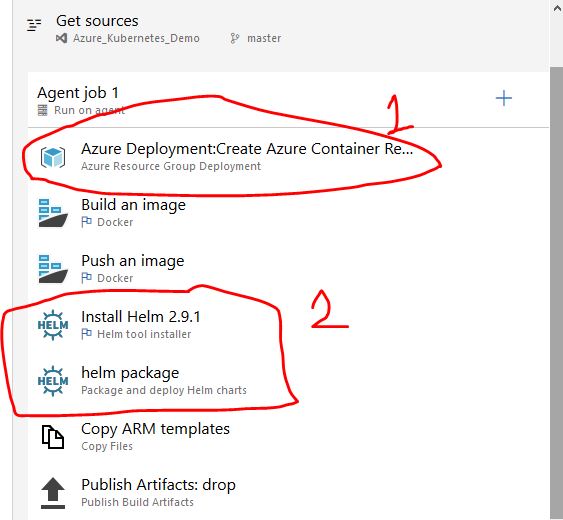
**Azure Kubernetes Demo**

* Import Repo from github (<https://github.com/KiranSonawane2003/Azure_Kubernetes_Demo.git>)



* Go to build create build by selecting template for *“Azure Kubernetes service”*  then add the following tasks



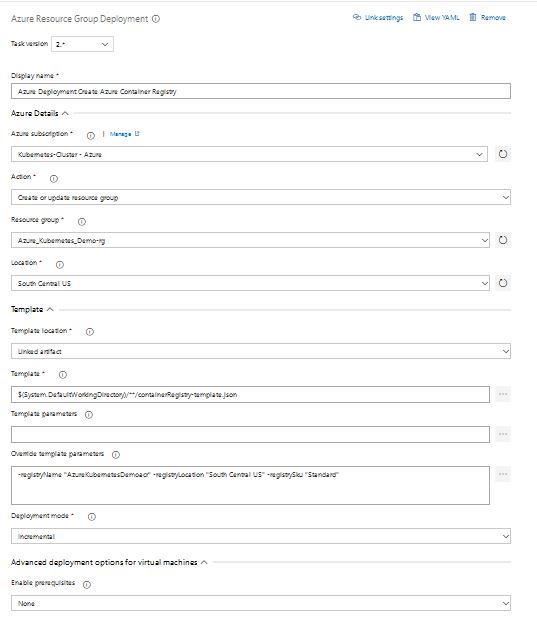
Here we use helm (package manage tool to manage Kubernetes)

Now we have to Change the each tasks setting as follow



C:\Users\ashishv\Desktop\Capture.JPG

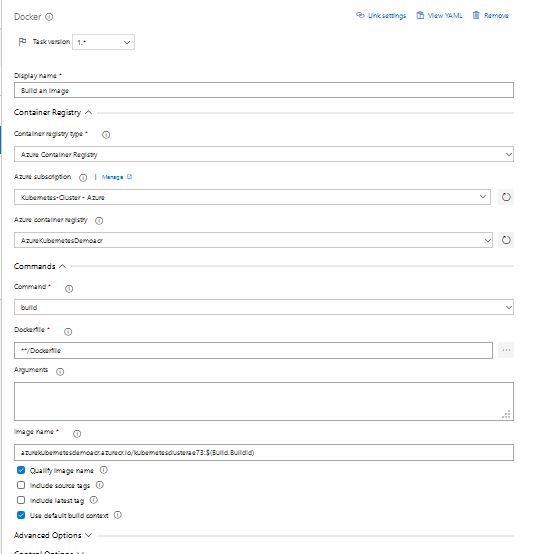
For Given task we have to set parameters as following screenshot





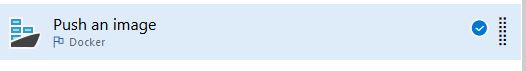
C:\Users\ashishv\Desktop\Capture.JPG

For building Nodejs docker image we have to set following settings

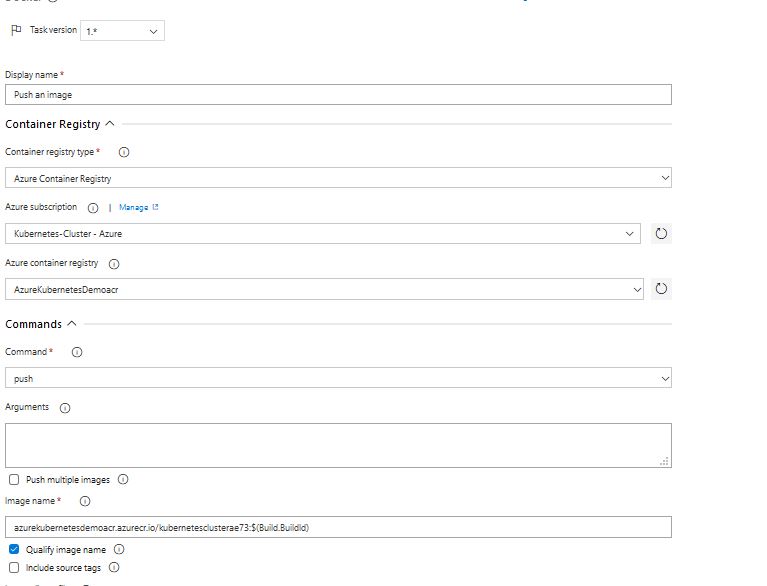


Have to Select Azure Credentials and Give the azure registry (We have to create Registry in azure before selecting registry name).

1. To push just created docker image to azure registry service following task

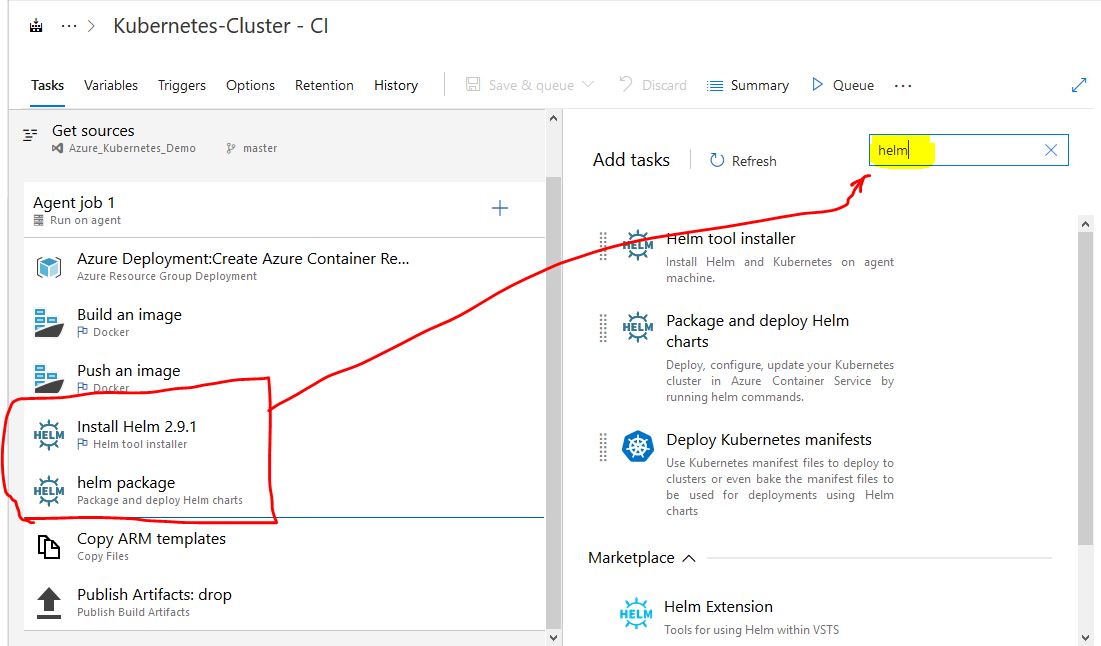


Changes to made in the task are

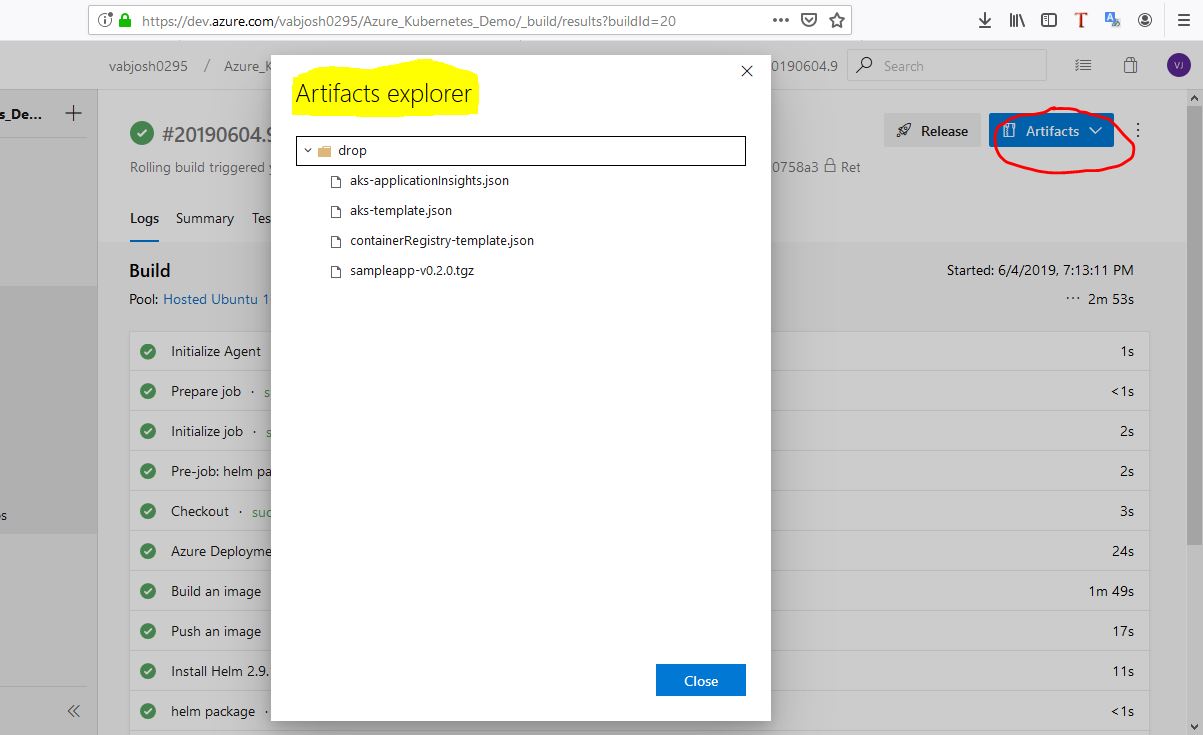


Have to Select Azure Credentials and Give the azure registry (We have to create Registry in azure before selecting registry name.) name where you want to upload your docker images which we just created.

1. Then just add Helm task so it can create all the resource communication for Kubernetes deployment. (Only if it’s not already added. Sometime we have to download it from marketplace)

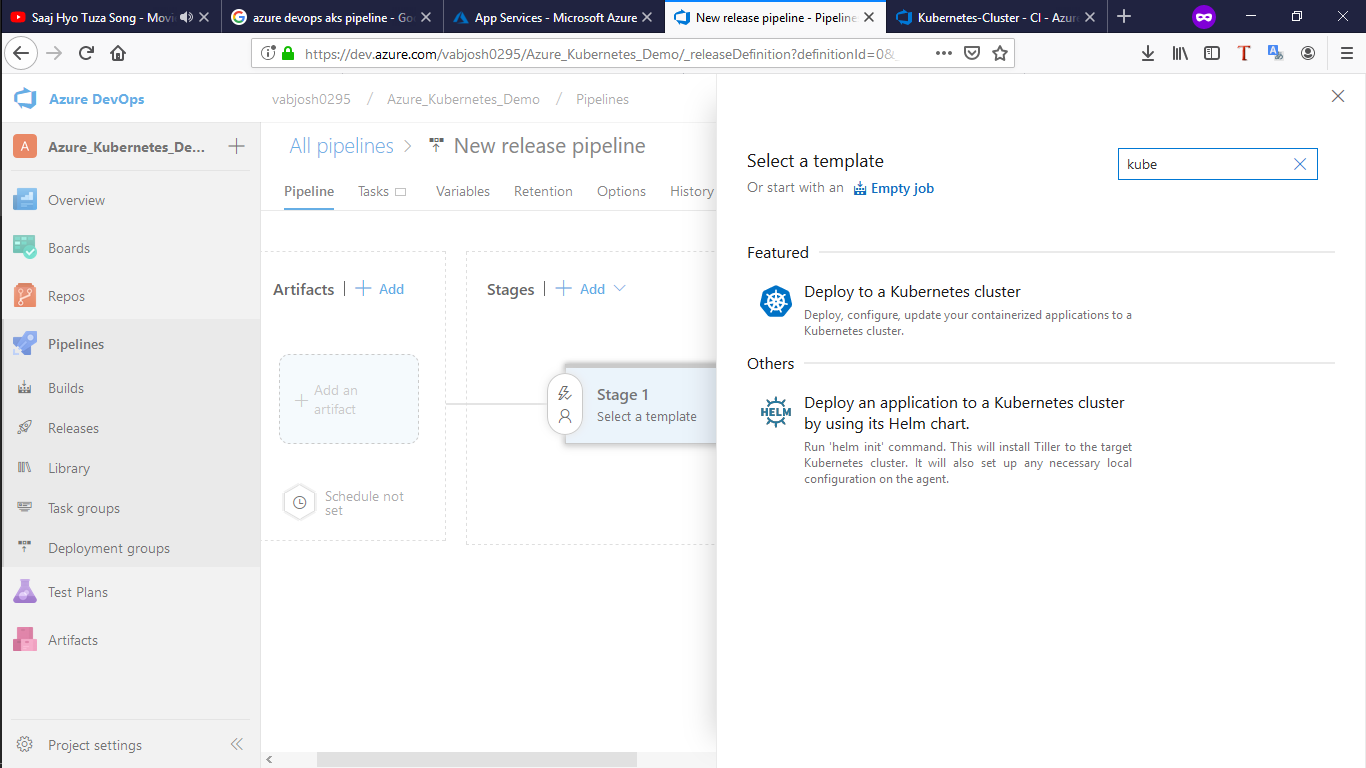


* Now build the ***“Build Pipeline”***

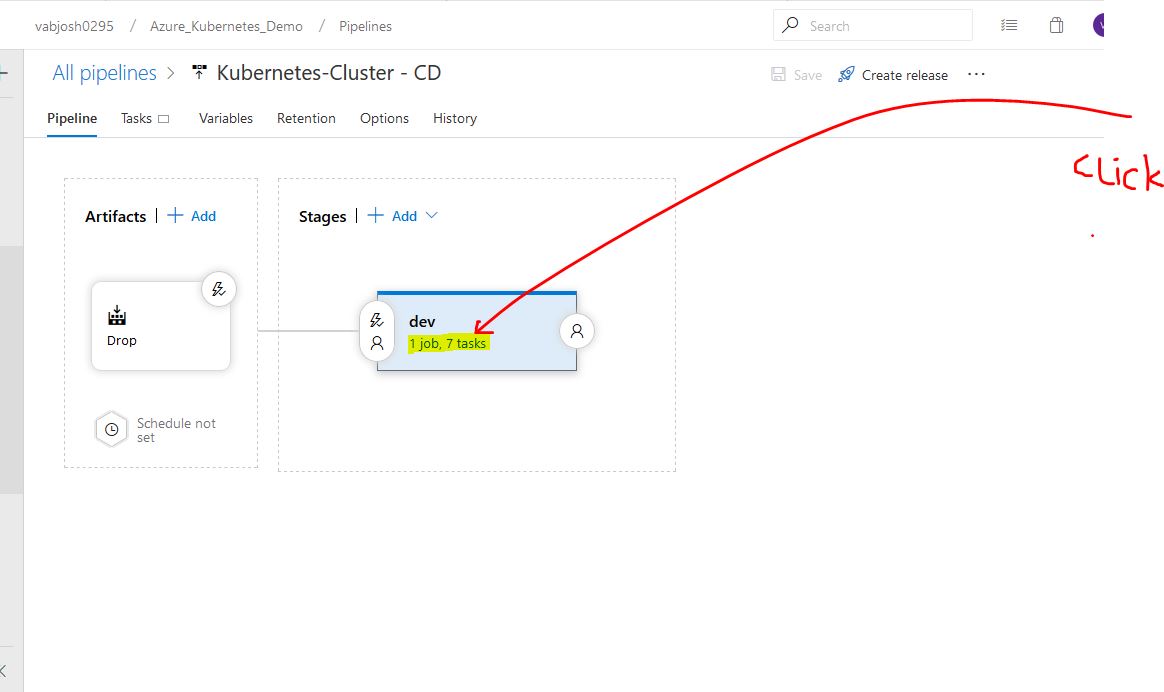


Once the build is success we can check the artifacts by clicking on Artifact button (Right side upper corner).

* Now our Artifact is created we can release the app to AKS.
* To release the app select the release button (beside Artifact)
* Then select Deploy an Application… using Helm

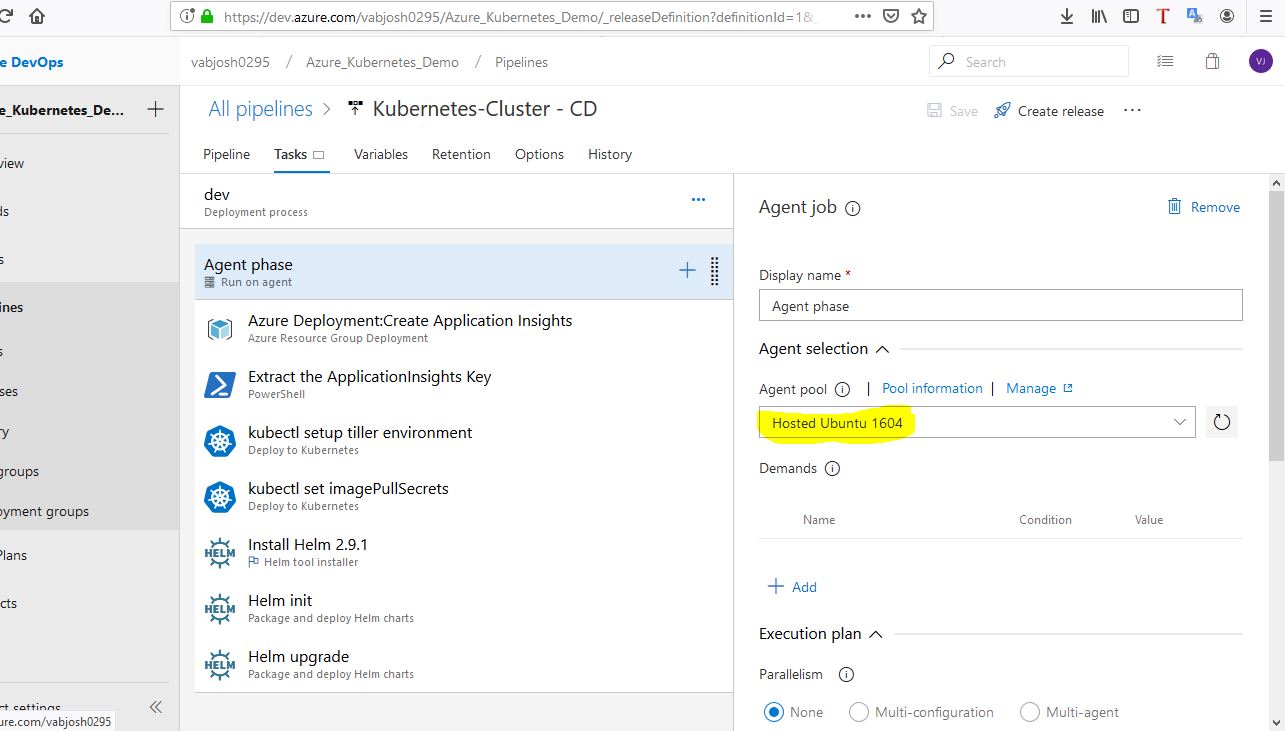


(Deploy an Application to Kubernetes cluster by using Helm Chart)



Select the shown line to enter the release task for setting.

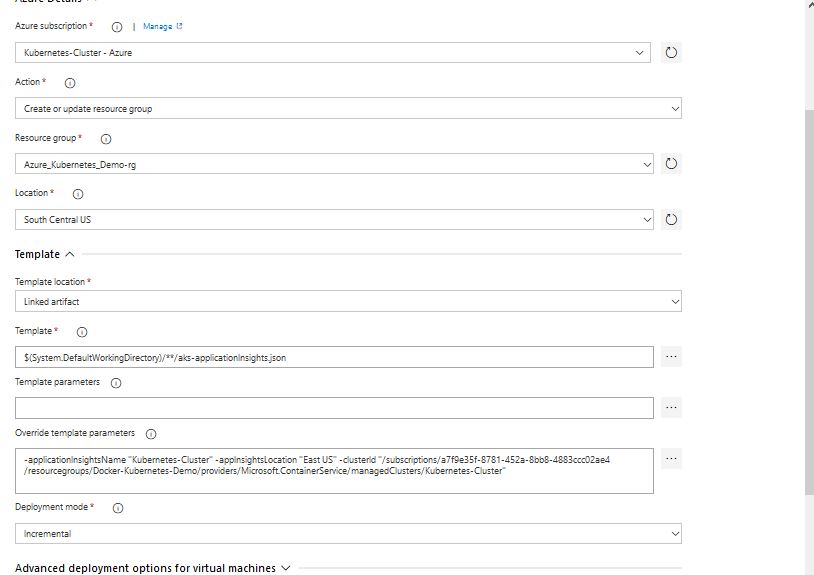
* Select the agent which should be ubuntu



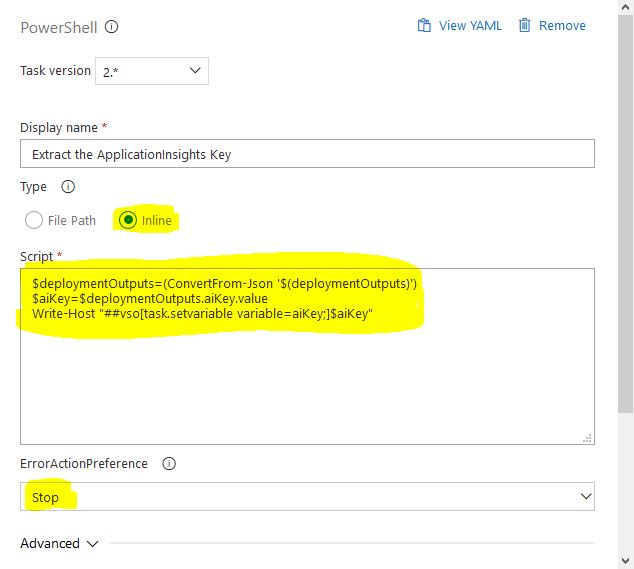
* Select the first task

C:\Users\ashishv\Desktop\Capture.JPG

Do the setting as follow



* Select the second task (i.e. powershell inline)and select inline option (so we can write shell commands to execute one by one in powershell)

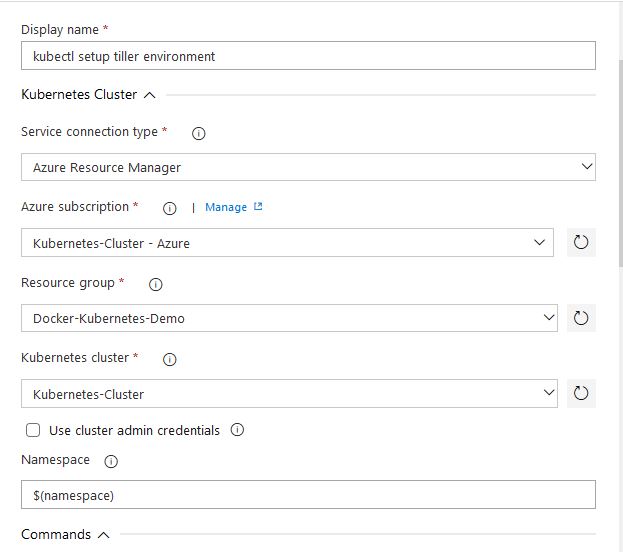


* Select the next task

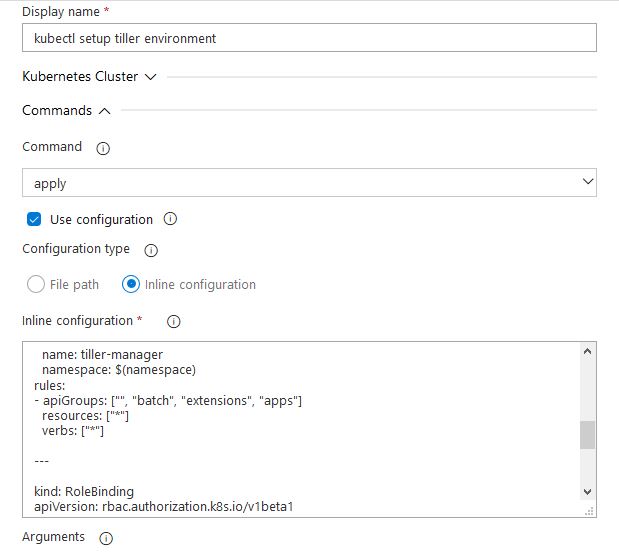
C:\Users\ashishv\Desktop\Capture.JPG

(This is called kubecuddel )

Now fill the Azure resource as required in following



Now pass the arguments as follow in ***Command***



Paste the following Script in ***Inline Configuration***

kind: Namespace

apiVersion: v1

metadata:

name: $(namespace)

spec: {}

status: {}

---

kind: ServiceAccount

apiVersion: v1

metadata:

name: tiller

namespace: $(namespace)

---

kind: Role

apiVersion: rbac.authorization.k8s.io/v1beta1

metadata:

name: tiller-manager

namespace: $(namespace)

rules:

- apiGroups: ["", "batch", "extensions", "apps"]

resources: ["\*"]

verbs: ["\*"]

---

kind: RoleBinding

apiVersion: rbac.authorization.k8s.io/v1beta1

metadata:

name: tiller-binding

namespace: $(namespace)

subjects:

- kind: ServiceAccount

name: tiller

namespace: $(namespace)

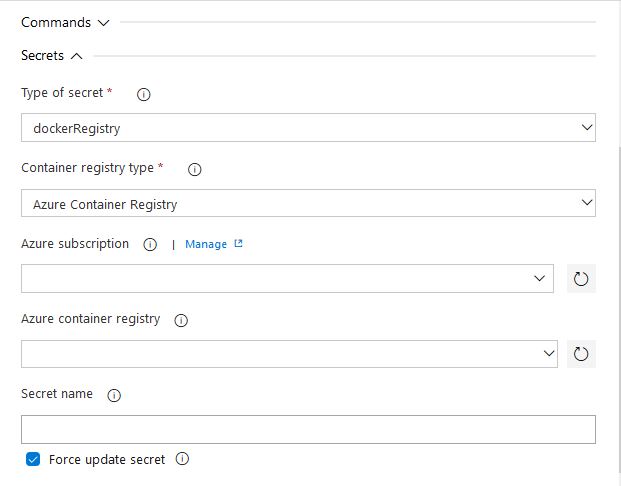
roleRef:

kind: Role

name: tiller-manager

apiGroup: rbac.authorization.k8s.io

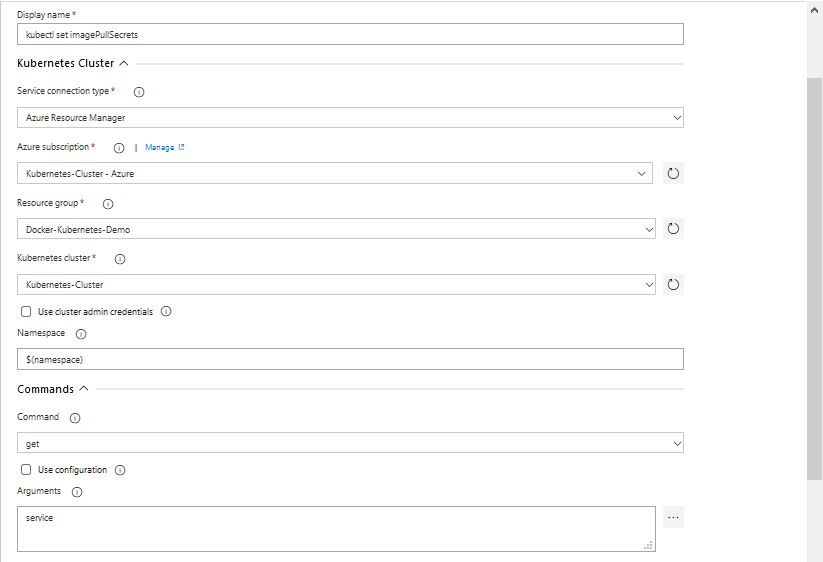
Now select ***Secretes*** for setting



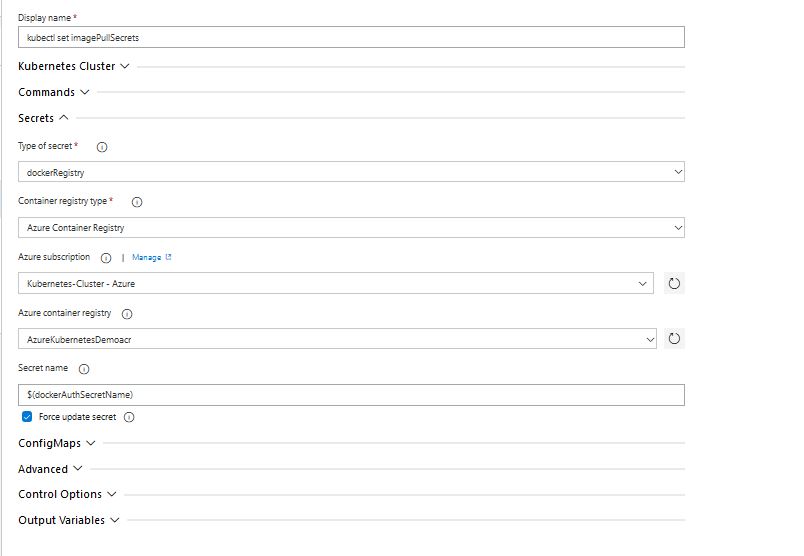
* Select the next task

C:\Users\ashishv\Desktop\Capture.JPG

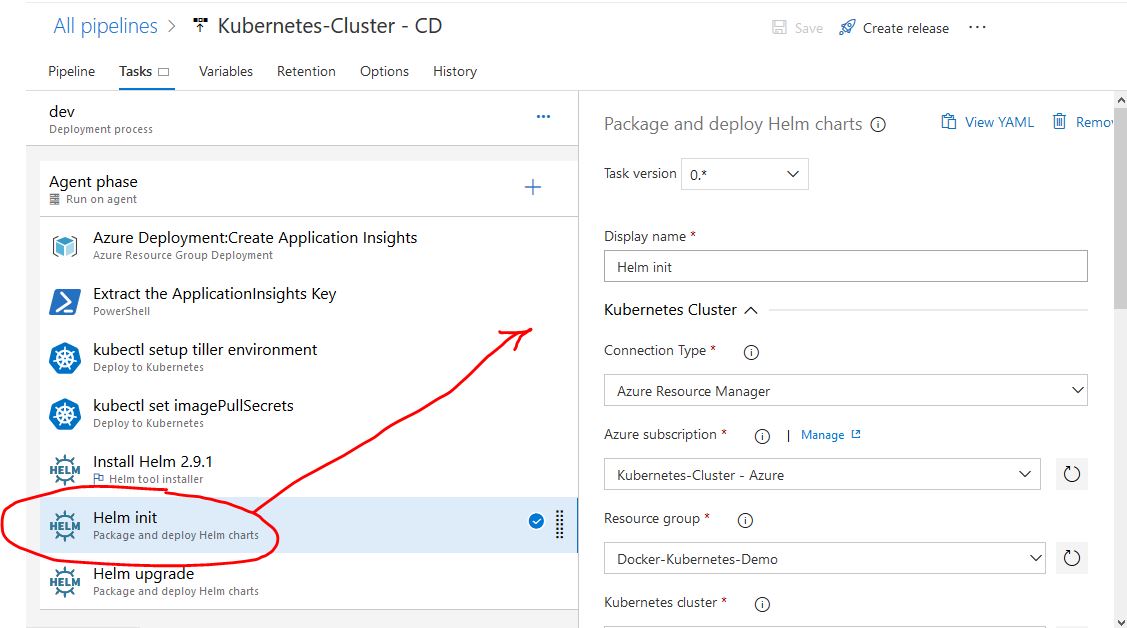
Do the following changes



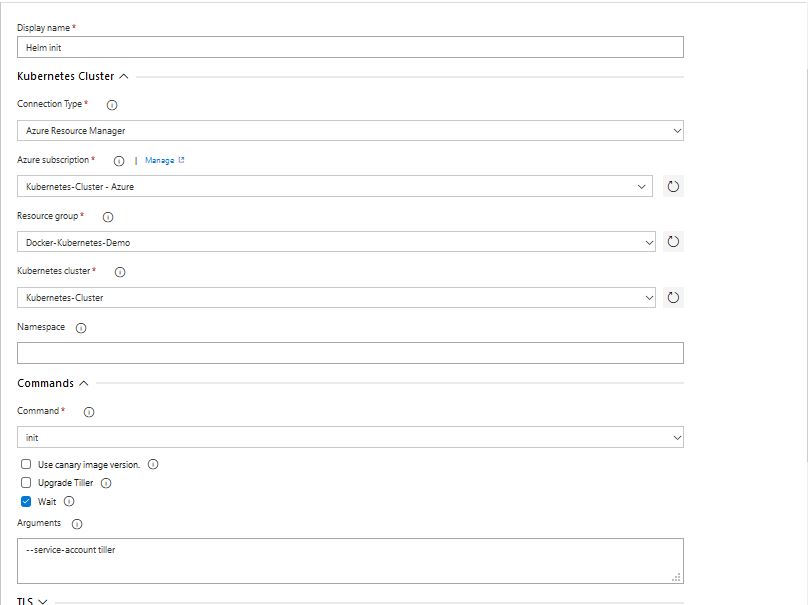
Select the ***Secretes*** and make following changes



* Select the following task for the setting

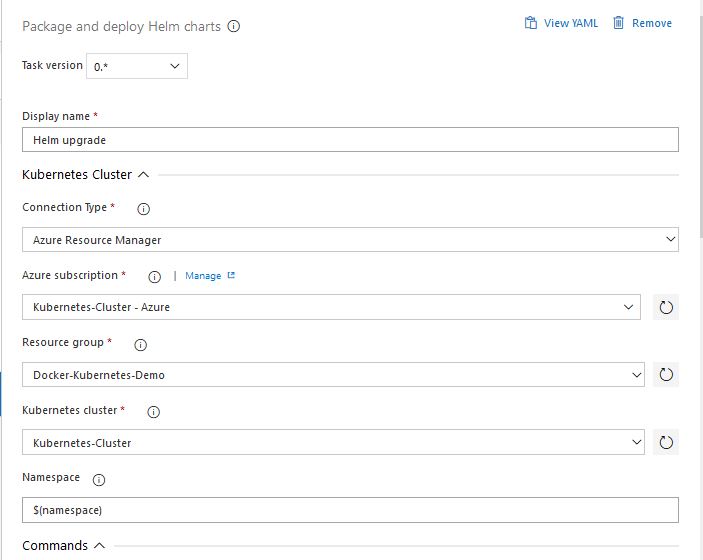


Following are the setting we need to change

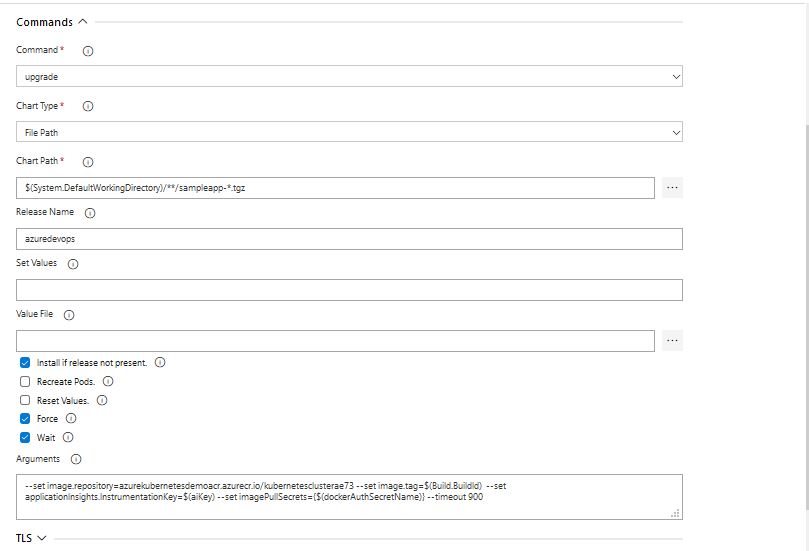


* Select the next task





Then make changes in commands



Now save and release the pipeline

And check the azure cluster for nodes.