**Kubernetes Cluster Setup using Azure Container Service**

(\*Note- If these brackets “<>” appear somewhere then replace it with proper values without putting <>)

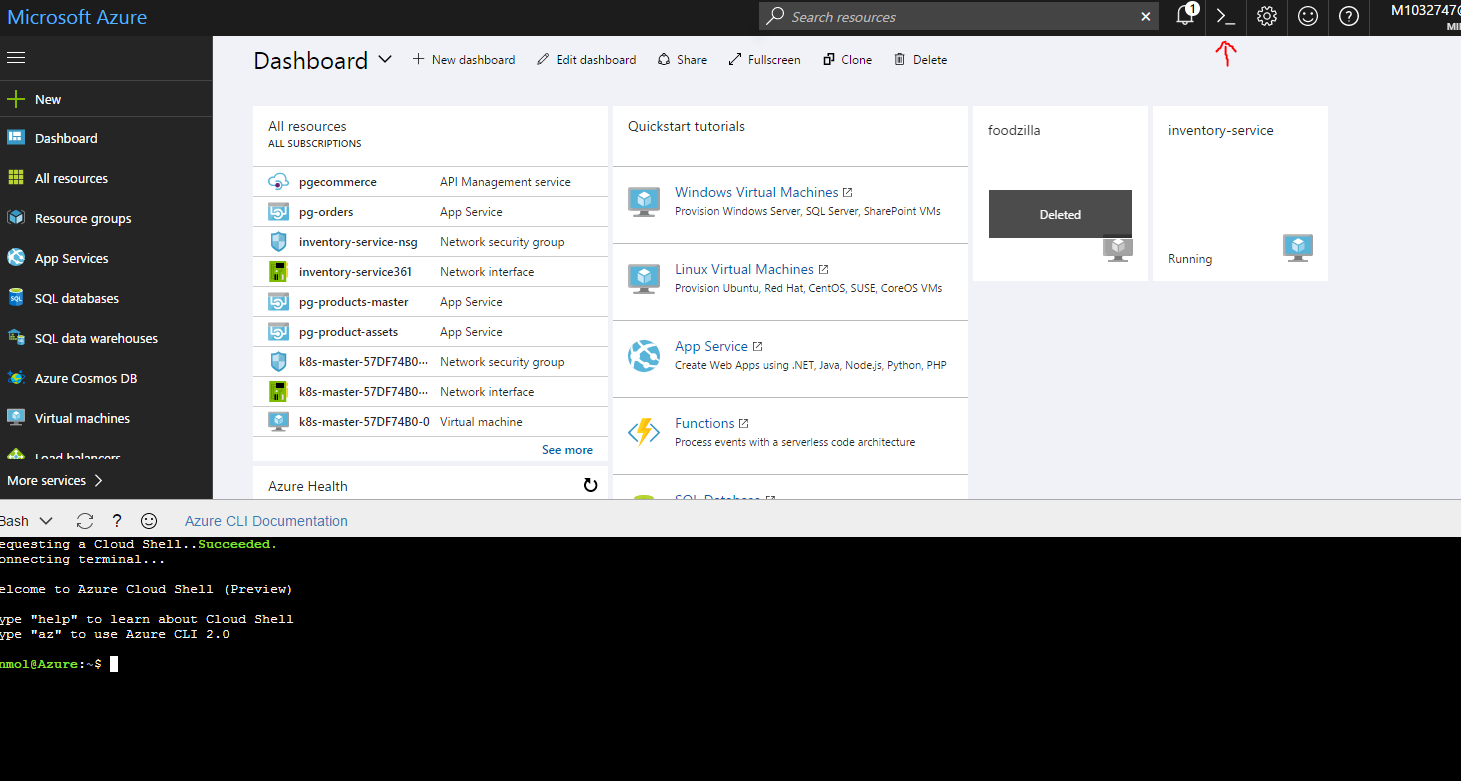
**Prerequisites:**

1. Azure Account Subscription
2. Azure CLI 2.0 – either via portal or download the Azure CLI 2.0.

**Steps:**

1. Ignore the steps 2- 8 and directly go to step 9 if joining the existing cluster and not planning to create a new cluster.
2. **Azure Portal Login**

Login to Azure Portal with your credentials and click on Azure CLI. The CLI will appear as shown in the figure.



1. **Creation of Resource Group**

Write these commands in Azure CLI

**RESOURCE\_GROUP= <my-resource-group>**

**LOCATION=westus** (Or any other)

**az group create --name=$RESOURCE\_GROUP --location=$LOCATION**

1. **To add DNS PREFIX and CLUSTER NAME**

Write these commands:

**DNS\_PREFIX=<some-unique-value>** (It will appear in DNS of Master, give relevant alphanumeric name)

**CLUSTER\_NAME=<any-acs-cluster-name>** (Give a proper name as it will be used later on)

1. **Starting the Cluster through ACS**

Write this command next in the CLI

**az acs create**

**--orchestrator-type=kubernetes**

**--resource-group <my-resource-group>**

**--name=$CLUSTER\_NAME**

**--dns-prefix=$DNS\_PREFIX**

**--generate-ssh-keys**

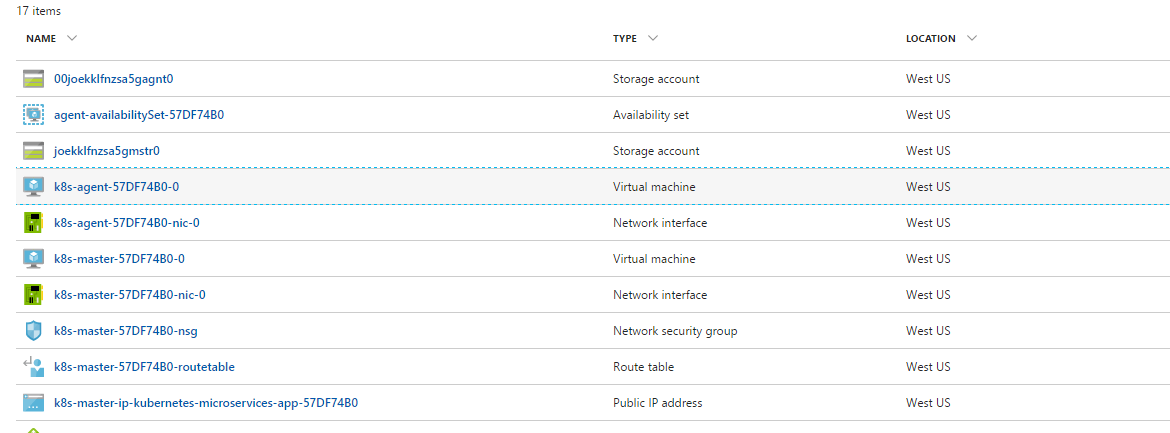
**--admin-username=<username>** (Any username you can remember)

**--admin-password=<password>** (Any password easy to remember)

**--master-count=1**

**--agent-count=1**

After few mins of command completion, a cluster would be up and running under the resource group created. Clicking on the resource group, observe few items like the ones given below.



1. **To Update Configuration of the Cluster**

Run command:

**az acs kubernetes get-credentials --resource-group=<my-resource-group> --name=$CLUSTER\_NAME**

This command will download the Kubernetes cluster Configuration to the **~/.kube/config** file.

(*This is very important command and shouldn’t be missed otherwise cluster won’t be setup properly*)

1. **Download Kubectl in local machine**

Now after connecting to cluster, in order to view the Kubernetes Dashboard, it’s recommended to see the steps for installing Kubectl library in your local machine. (<https://kubernetes.io/docs/tasks/tools/install-kubectl/> )

1. **Configure Kubectl and setup cluster configuration in local**

To configure kubectl, first copy the content from **~/.kube/config** file in the AZURE CLI using **cat config** command. Then switch to **<C:\Users\<your username>\.kube>**. If .kube folder is not there at specified location, create one. Now create one file named **config** inside .**kube** directory (don’t give any extension to file). Paste the contents copied from azure cli to this file and save it. Append kubectl.exe directory path into your system path variable.

*(Note- Check the config file properly after copying as it is a yaml file where extra whitespaces and wrong indentation can cause issues, use* ***config*** *file added in the documentation repository for reference)*

1. **Connect to existing cluster**

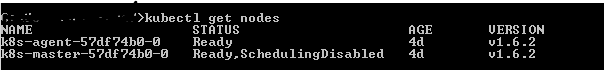
In order to connect to the existing cluster from your local or azure portal, copy and paste the config file from <https://github.com/handaanmol/documentation> to **<C:\Users\<your username>\.kube**> for windows or to **~/.kube/ folder for azure portal CLI** and run the kubectl commands from cmd/CLI. Follow steps 7 and 8 to setup and configure kubectl.

*(Note – This step is not valid for new cluster)*

1. **Check working of nodes**

Run command: **kubectl get nodes**

This will give you information about the master node and agent node running in the cluster



Upon execution of this command, observe the status of both nodes as ready which means the cluster is working fine.

1. **Connecting to Kubernetes Dashboard from local**

Run command: **kubectl proxy**

View the Kubernetes Dashboard by following the above command. You can view the dashboard on <http://localhost:8001/ui>

It will look like this

