**Kubernetes Cluster Setup using Azure Container Service**

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

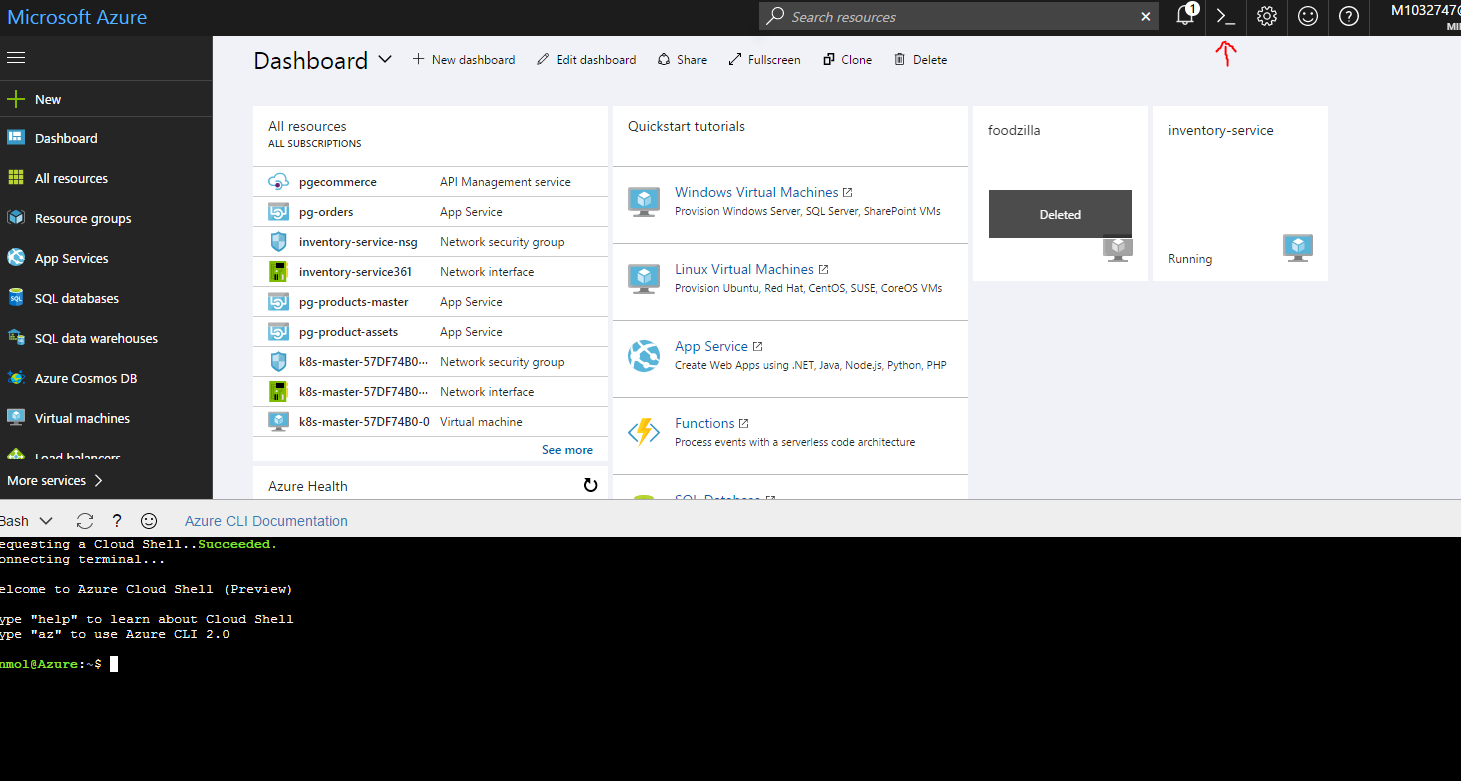
**Prerequisites:**

1. Azure Account Subscription
2. Azure CLI 2.0 (in your local system, if possible) Either portal Azure CLI or download the Azure CLI 2.0.
3. Kubectl.exe file in your local machine (a package used to run kubernetes command) to see Kubernetes Dashboard once deployed on local as they don’t provide direct access to dashboard. Again its optional
4. SSH Keys – You can create them initially and store them in ~/.ssh/id\_rsa.pub file or you can create them later during cluster creation.

**Steps:**

1. If joining the existing cluster and not planning to create a new cluster, kindly ignore the steps 2- 8 and directly go on step 9.
2. **Azure Portal Login**

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

(this command creates a resource group, same can be done from UI also)

1. **To add a DNS\_PREFIX and CLUSTER\_NAME**

Write these commands :

**DNS\_PREFIX=<some-unique-value>**  (it will appear in DNS of Master, give alphanumeric names unique to what you are doing)

**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**

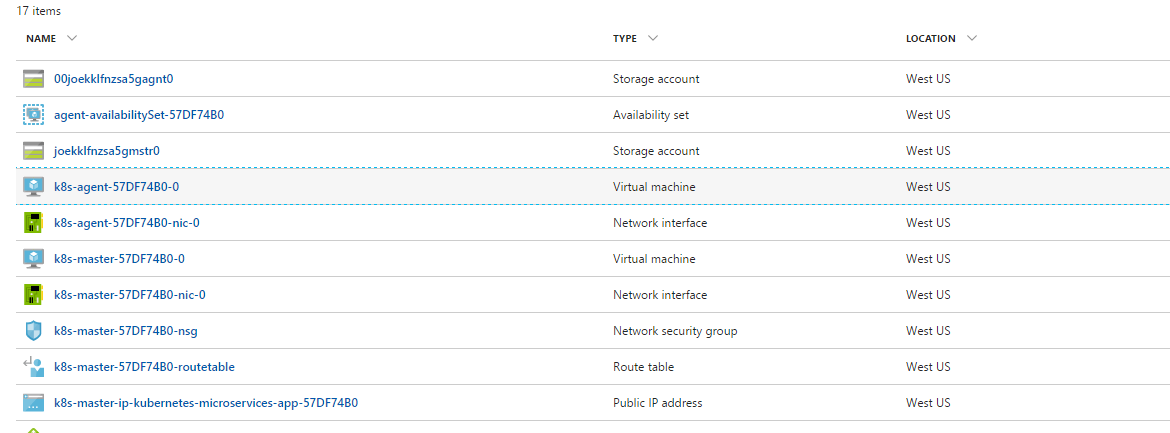
**--admin-username=<username> (Any username you can remember)**

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

**--master-count=1**

**--agent-count=1**

After few mins when this command completes, note that a cluster has been setup under the resource group created. Clicking on the resource group, observe few items like the ones given below with some of them having names starting with k8.



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

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 download 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 config file inside .**kube** directory (don’t give any extension to file). Paste the contents copied from azure cli to this file and save it. Now, append kubectl.exe directory path in 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, for reference, use config file added in the documentation repository)*

1. **Connect to existing cluster**

In order to connect to the existing cluster from your local or azure portal which is already up and running , copy and paste the config file attached in this documentation to **<C:\Users\<your username>\.kube**> for windows or in **~/.kube/ folder for azure portal CLI** and run the kubectl commands from cmd/CLI.

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

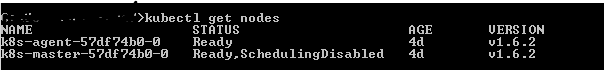
(*Create .****kube*** *folder if doesn’t exist using cmd command* ***mkdir .kube*** )

Refer to this link for installing kubectl: (<https://kubernetes.io/docs/tasks/tools/install-kubectl/> )

1. **Check working of nodes**

Command: **kubectl get nodes**

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



Once you execute this and see the status of both nodes as ready, that means your cluster is working fine.

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

Use the command **kubectl proxy** to view the Kubernetes Dashboard by setting up a proxy tunnel to the Kubenetes API Server. You can view the dashboard on <http://localhost:8001/ui>

It will look like this

