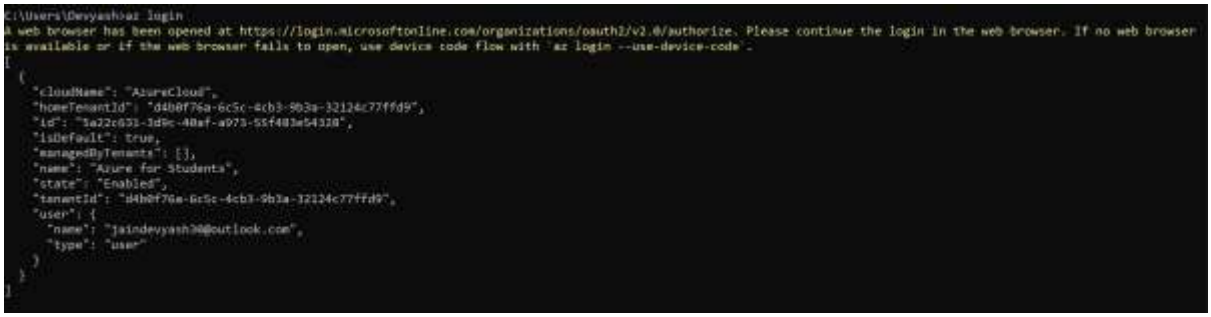
Steps to Implement

1) Set up Azure CLI

Make sure you have the Azure CLI installed on your local machine. If not, you can install it by following the instructions on the Azure documentation.

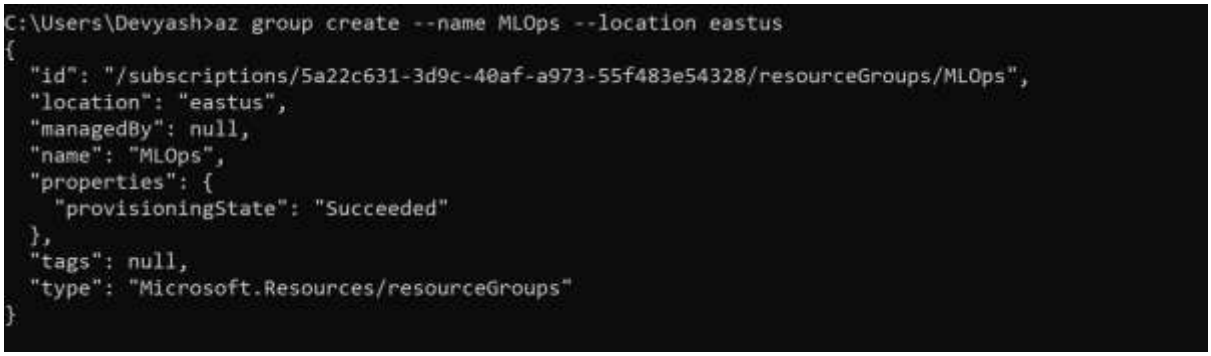
2) Login to Azure

Log in to your Azure account using the CLI:



3) Create a Resource Group

Create a new resource group in a location of your choice:



4) Create an AKS Cluster

Create an Azure Kubernetes Service cluster:

5) Configure kubectl

Configure kubectl to connect to your new AKS cluster:

Code: ‘az aks get-credentials --resource-group MLOPs --name MLOPs’

6) Install Kubeflow

To install Kubeflow, you'll need to use kfctl, the Kubeflow command-line interface.

Code:

export KF_NAME=TEE

export BASE_DIR=path to directory

export KF_DIR=${BASE_DIR}/${KF_NAME}

Use kfctl to apply the configuration:

kfctl apply -V -f ${CONFIG_URI}

7) Access Kubeflow Dashboard

After applying the configuration, wait for the services to come up. Then you can access the Kubeflow dashboard by forwarding the istio-ingressgateway service port to your local machine:

Code: ‘kubectl port-forward svc/istio-ingressgateway -n istio-system 8080:80’

8) Set Up Storage and Other Services

Depending on your needs, you might have to set up persistent storage, databases, or other services. Azure offers integrated services like Azure Blob Storage, Azure File Storage, Cosmos DB, etc., that can be used with AKS.

9) Monitor and Manage your Kubeflow Cluster

Utilize Azure's monitoring tools to keep an eye on your cluster's health and performance. You can also set up alerts and automated scaling based on metrics.

10) Deploy and Use Kubeflow Pipelines

Create and deploy your machine learning pipelines using the Kubeflow Pipelines SDK. You can manage your ML workflows through the Kubeflow dashboard.