Create and deploy .NET Core App for Azure Kubernetes

Hamida REBAI

Microsoft MVP & MCT

Microsoft MVP in Developer Technologies

Member and Speaker at dotnetfoundation

Blogger and Technical writer

Rebai Hamida – Medium

Hamida Rebai Trabelsi | LinkedIn

Rebaï Hamida - YouTube

Microsoft MVP

MVP

Microsoft Most Valuable Professional

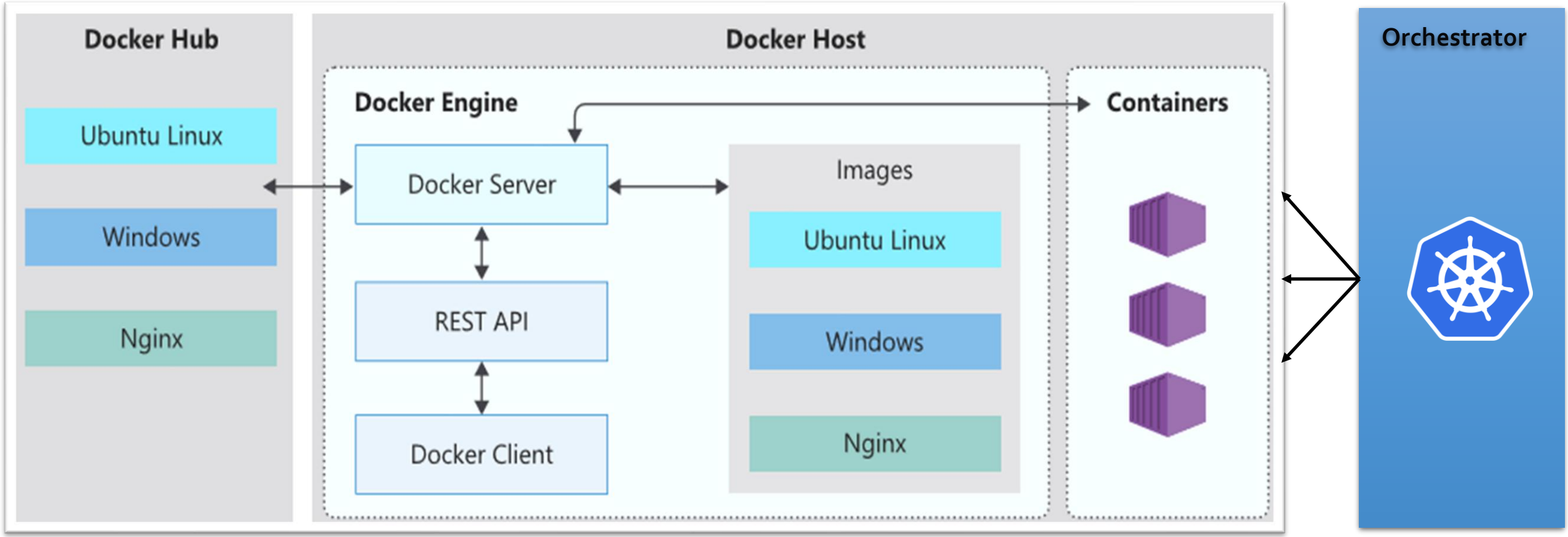
AGENDA

- Kubernetes
- Kubernetes vs. Docker
- Azure Kubernetes Services
- Prerequisite
- Demo: Create Kubernetes application in Visual Studio 2019
- Demo: Deploy the application on Azure Container Registry
- Demo: Create AKS cluster using Azure CLI
- Demo: Create AKS Cluster using Azure Portal
- Demo: Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI
- Deploy an Azure Kubernetes Service (AKS) cluster using an ARM template
- Deploy an Azure Kubernetes Service (AKS) cluster using Azure DevOps Starter
- Demo: Debug your application using Bridge to Kubernetes

Kubernetes

- Open-source container orchestration plateforme.
- Enables the operation of an elastic web server framework for cloud applications.
- Support data center outsourcing to public cloud service providers or can be used for web hosting at scale.

Kubernetes vs. Docker



Azure Kubernetes Services

- Azure Kubernetes Service (AKS) is a managed Kubernetes service that lets you quickly deploy and manage clusters.
- AKS handles critical tasks
- AKS is free and you pay only for the agent nodes (in the cluster not for the masters).

Azure Kubernetes Service Features

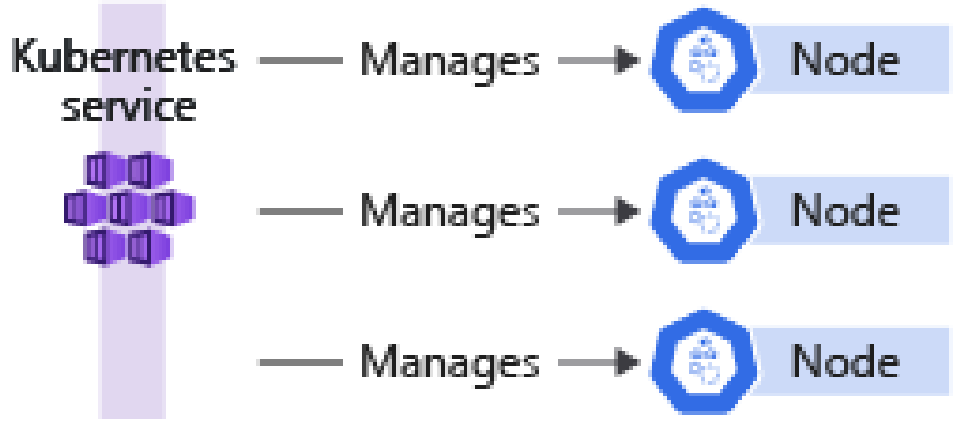
- Security, Access, and Monitoring
  - Identity and security management
  - Integrated logging and monitoring
- Clusters and Nodes
  - Cluster node and pod scaling
  - Cluster node upgrades
- Development Tooling Integration

Why we use AKS?

- Easy migration of the existing application to containers.
- Simple deployment and management of applications based on microservices.
- DevOps: faster delivery with Azure pipeline.
- Constant monitoring.
- Easy scaling.

Architecture Pattern for AKS

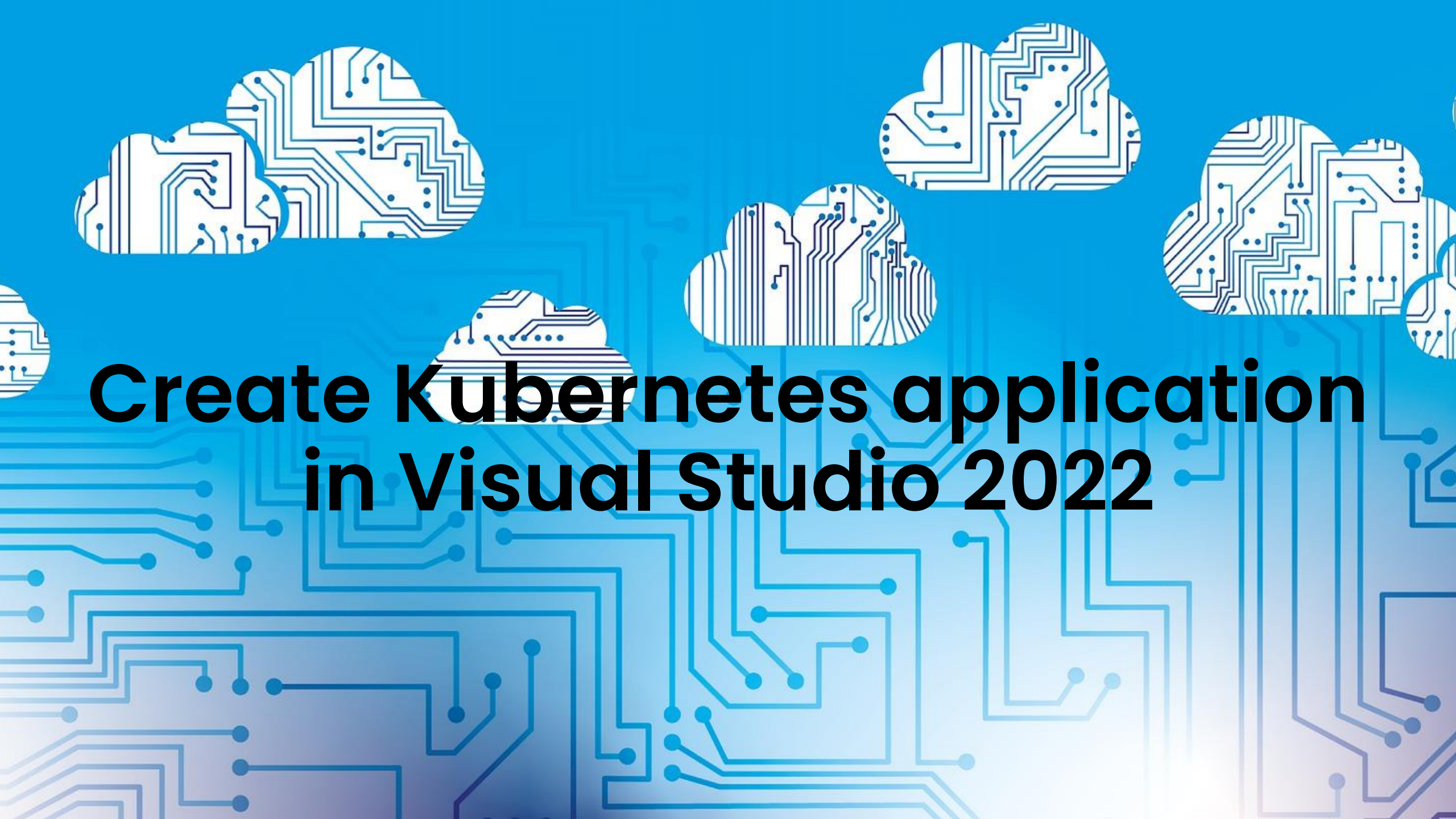
- Single control plane and multiple nodes



- Single control plane and a single node

Prerequisite

- Azure account: you should be able to create an AKS Cluster.
- Docker Community Edition
- Use the Bash environment in Azure Cloud Shell.
- Visual Studio 2019


Create Kubernetes application
in Visual Studio 2022

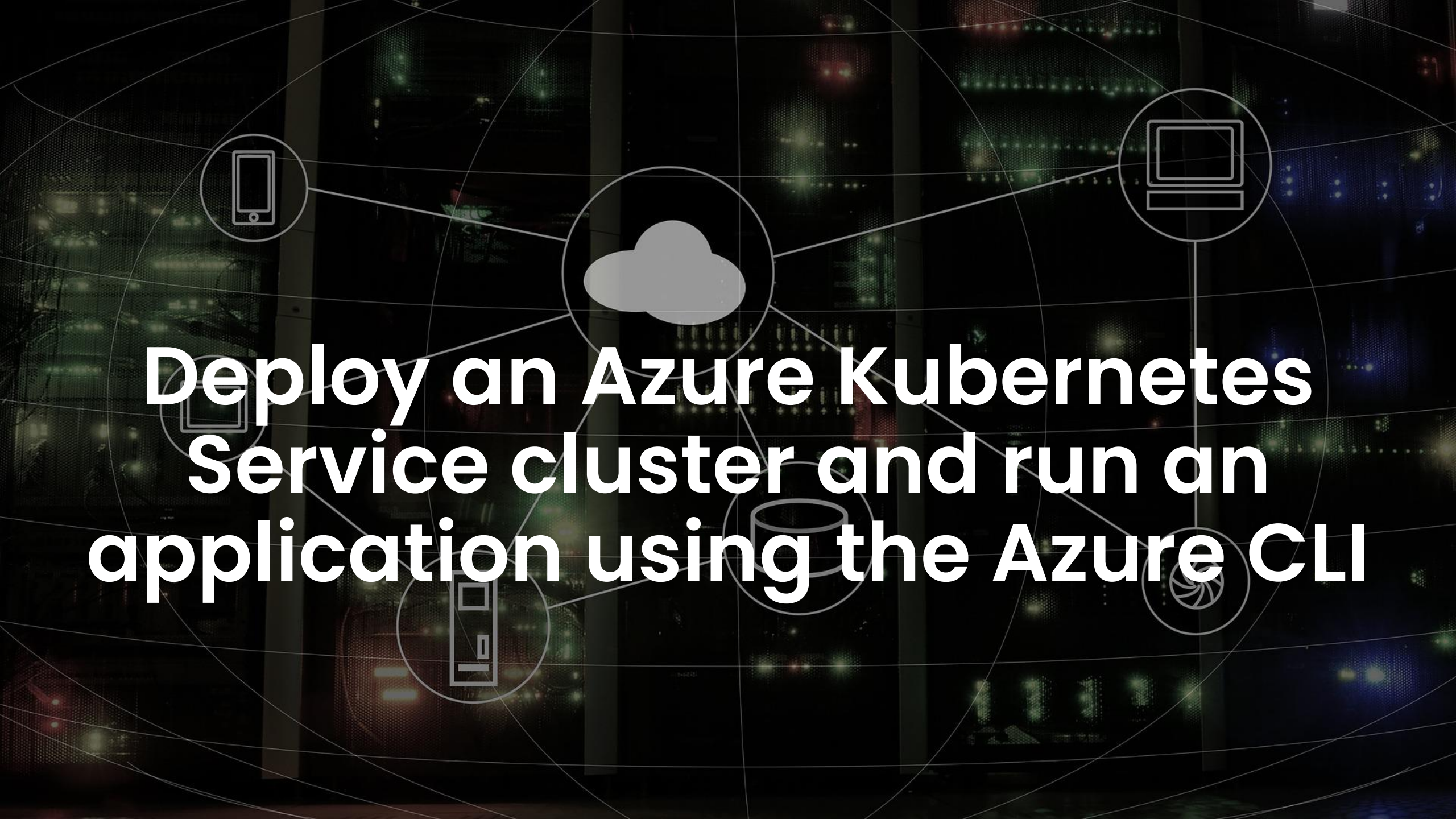
Deploy the application on Azure Container Registry

Create AKS Cluster using Azure Portal


Create AKS cluster using Azure CLI

Create AKS cluster using Azure CLI

- **az aks create** --resource-group aksgr --name myAKSClusterName --node-count 1 -- generate-ssh-keys --attach-acr **yourContainerName**

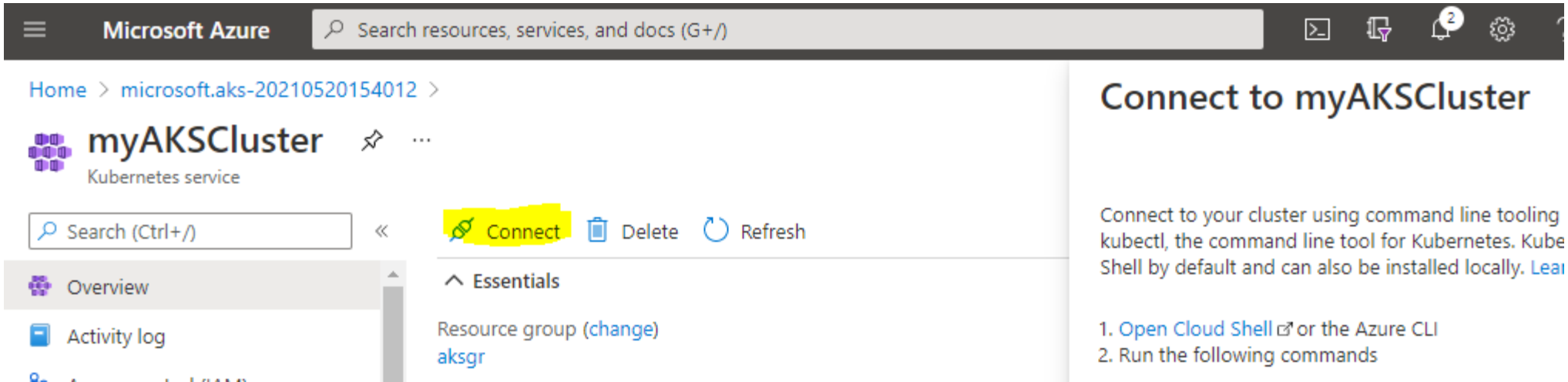


Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

- 1- Open CloudShell My Dashboard — Microsoft Azure and connect to your cluster as bellow:
- az account set — subscription yoursubscription
- az aks get-credentials — resource-group aksgr — name myAKSCluster

You can find these commands when you open Azure Portal and your cluster:



Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

- 2- Create an empty file called: azure-demo-deployment.yaml

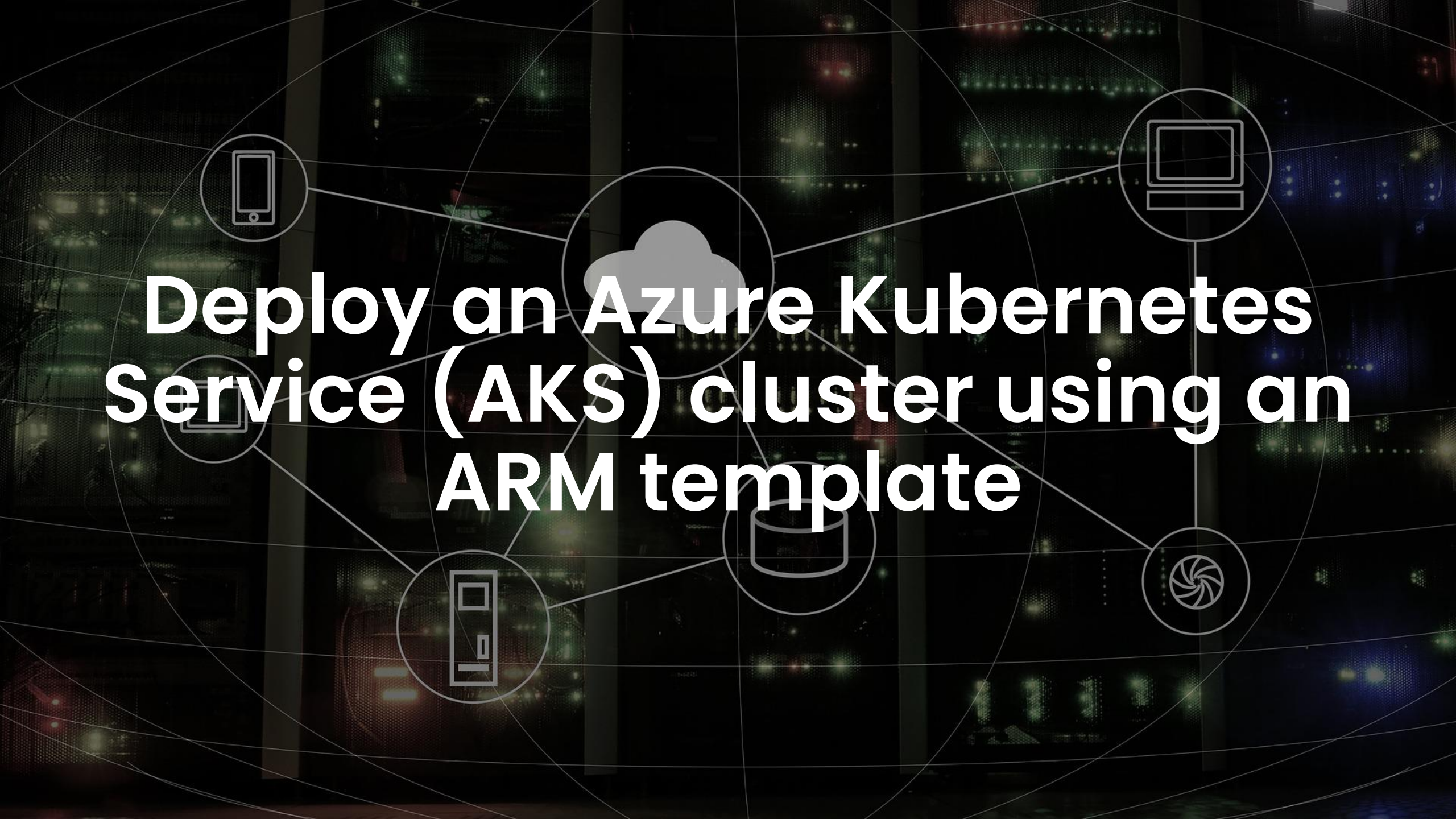
- 3- Copy this content to the empty file created, we will describe it after.

Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

```
apiVersion: apps/v1
kind: Deployment
metadata:
name: demo-kubernetes-deployment
spec:
selector:
matchLabels:
app: demo-kubernetes-pod
replicas: 1
template:
metadata:
labels:
app: demo-kubernetes-pod
spec:
containers:
-- name: yourContainerName
image: yourContainerName.azurecr.io/aksproject:latest
ports:
-- containerPort: 80
```

Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

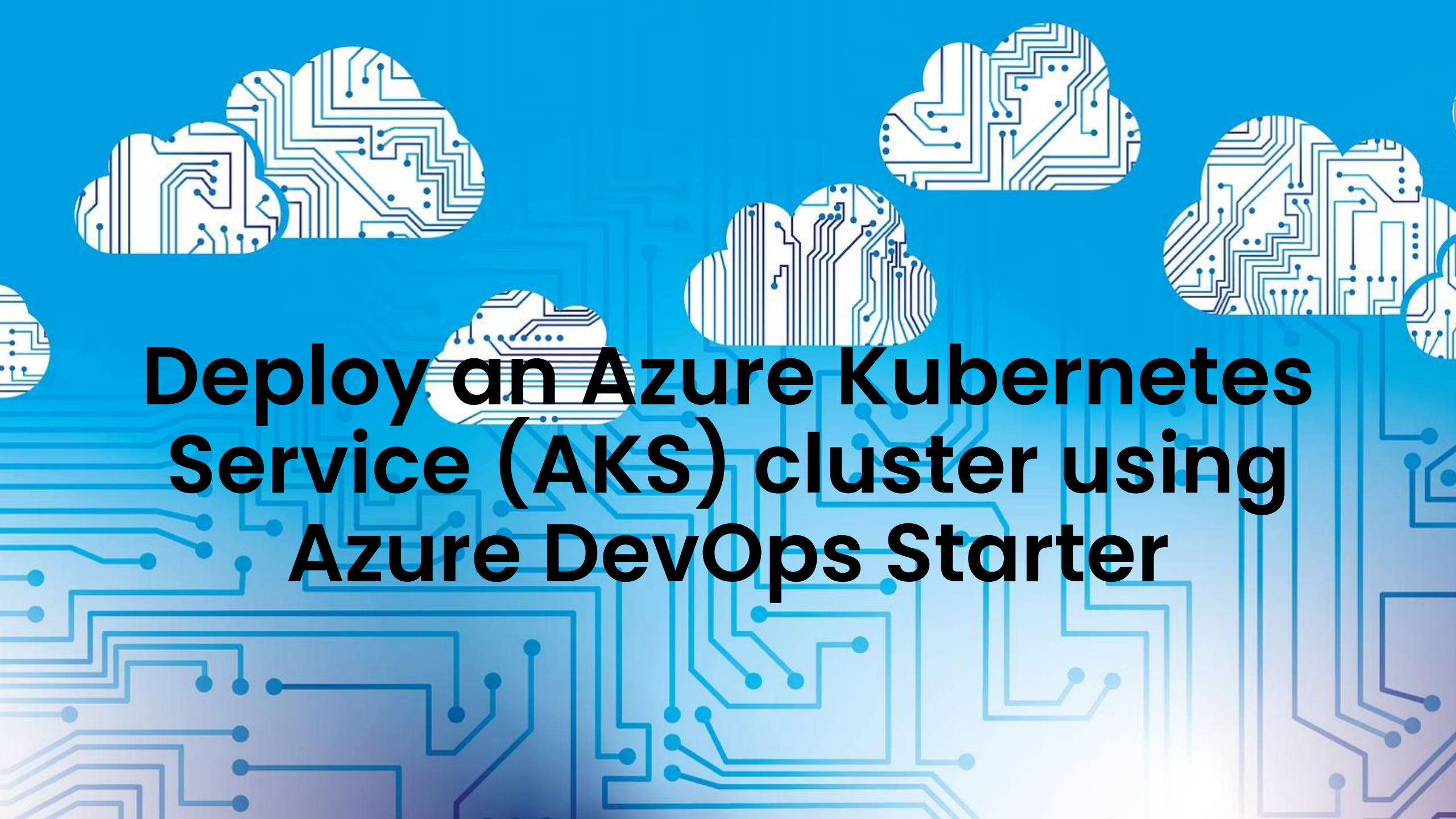
- 4- Run the application and deploy it in the cluster using the kubectl apply command and specify the name of your YAML manifest.
- **kubectl apply -f azure-demo-deployment.yaml**
- We will use **kubectl get deployments** to verify if the deployment was created or not.
- When the application runs, a Kubernetes service exposes the application front end to the internet. This process can take a few minutes to complete.
- 5- we will use **kubectl get service** command with the --watch argument.
- kubectl get service demo-kubernetes-deployment — watch

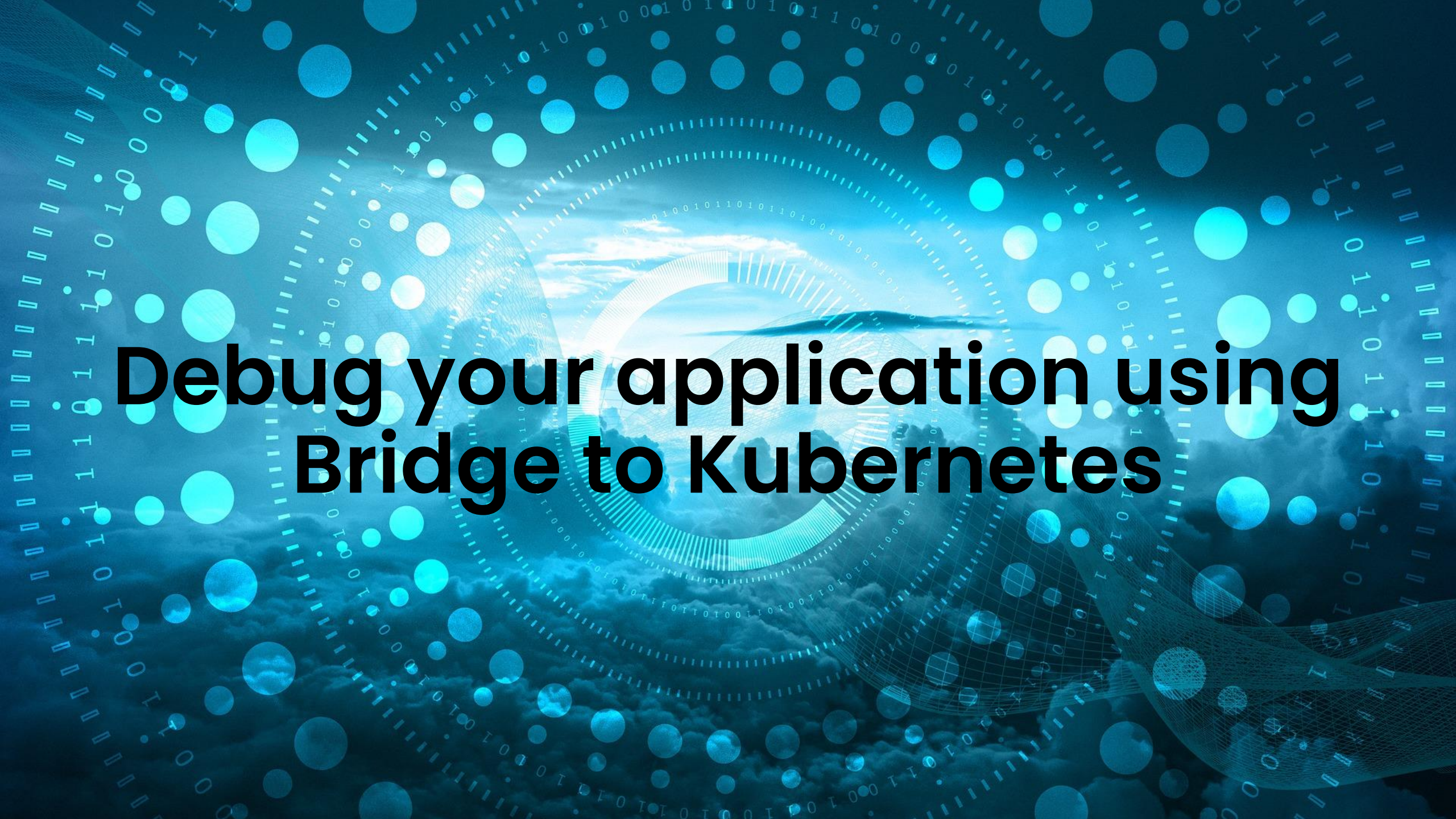


Deploy an Azure Kubernetes Service (AKS) cluster using an ARM template

ARM template

- JavaScript Object Notation (JSON) file
- Defines the infrastructure and configuration for your project.
- The template uses declarative syntax.
- In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.
- Link:


Deploy an Azure Kubernetes Service (AKS) cluster using Azure DevOps Starter


Debug your application using
Bridge to Kubernetes