

TECHNOLOGY



Container Orchestration using Kubernetes

Azure Kubernetes Service



A Day in the Life of a DevOps Engineer

You are working as a DevOps Engineer in an organization, and you have been asked to design a solution to deploy MySQL using the Azure dynamic storage class. You need to check the storage classes provided by the Azure Kubernetes Service (AKS) for deploying the MySQL Pods with Azure disk as Persistent Volume.

Once the Pods are deployed, you need to verify the Pod logs to ensure the MySQL is running without any issues.

To achieve all the above, along with some additional concepts, we would be learning a few concepts in this lesson that will help you find a solution for the above scenario.



Learning Objectives

By the end of this lesson, you will be able to:

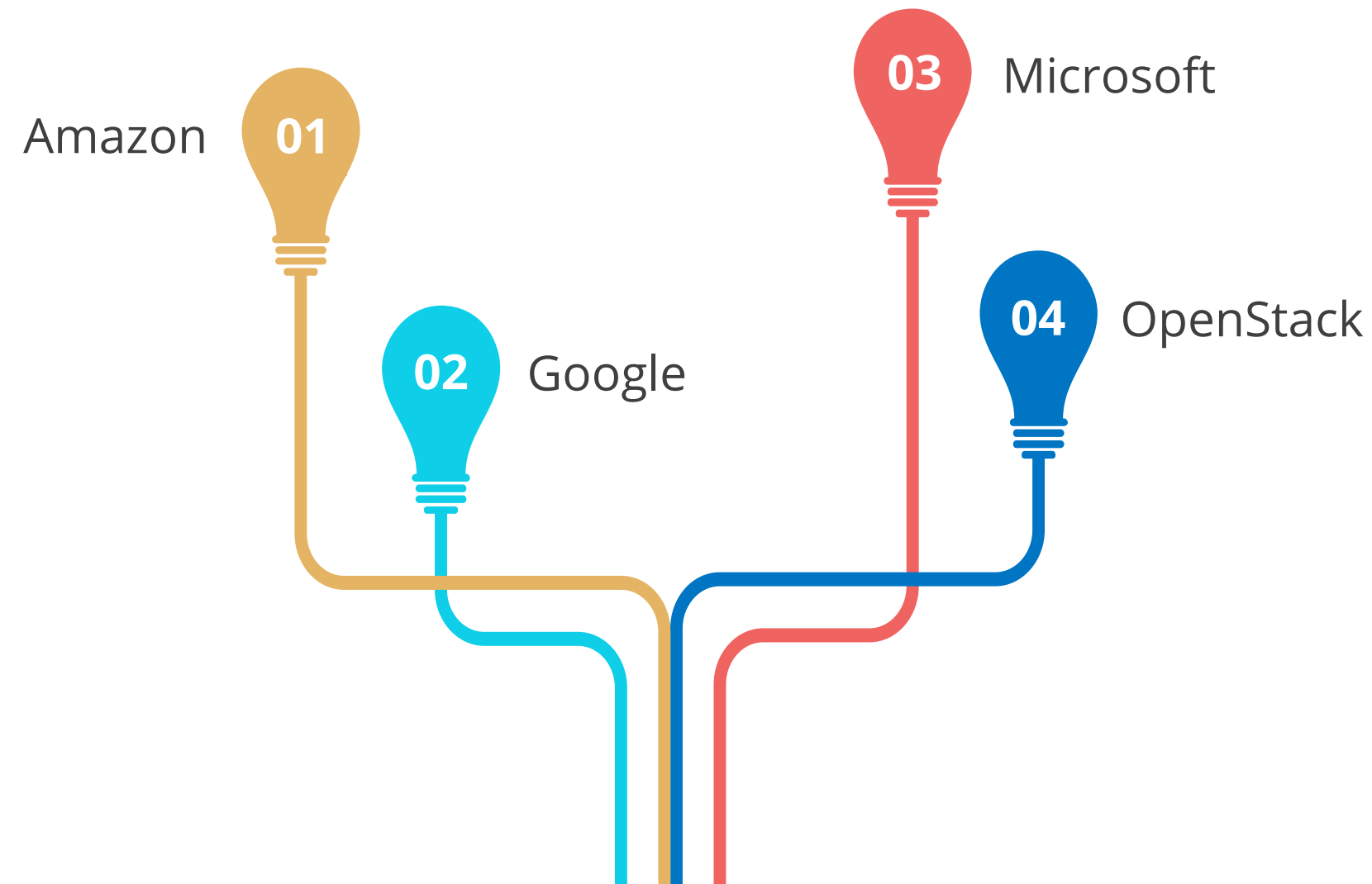
- 🕒 Create a Kubernetes cluster using AKS
- 🕒 Deploy Kubernetes workloads in an AKS cluster
- 🕒 Perform Pod scheduling and rollout update for workloads
- 🕒 Demonstrate persistent storage creation on the AKS cluster



Kubernetes in Cloud Environment

Kubernetes and Cloud Environment

Most cloud service providers have integrated Kubernetes as a part of the platform for their clients.
Some of the most important and influential providers are:



Kubernetes as a Service



Kubernetes as a Service (KaaS) assists users in migrating to cloud-native enabled Kubernetes-based platforms and managing the lifetime of K8s clusters.

Kubernetes as a Service (KaaS)

KaaS provides customizable solutions based on an organization's existing requirements and data center scale while keeping budget restrictions in mind. Some of the advantages of KaaS are:

1

Security

2

Saving in investment for resources

3

Scaling of infrastructure



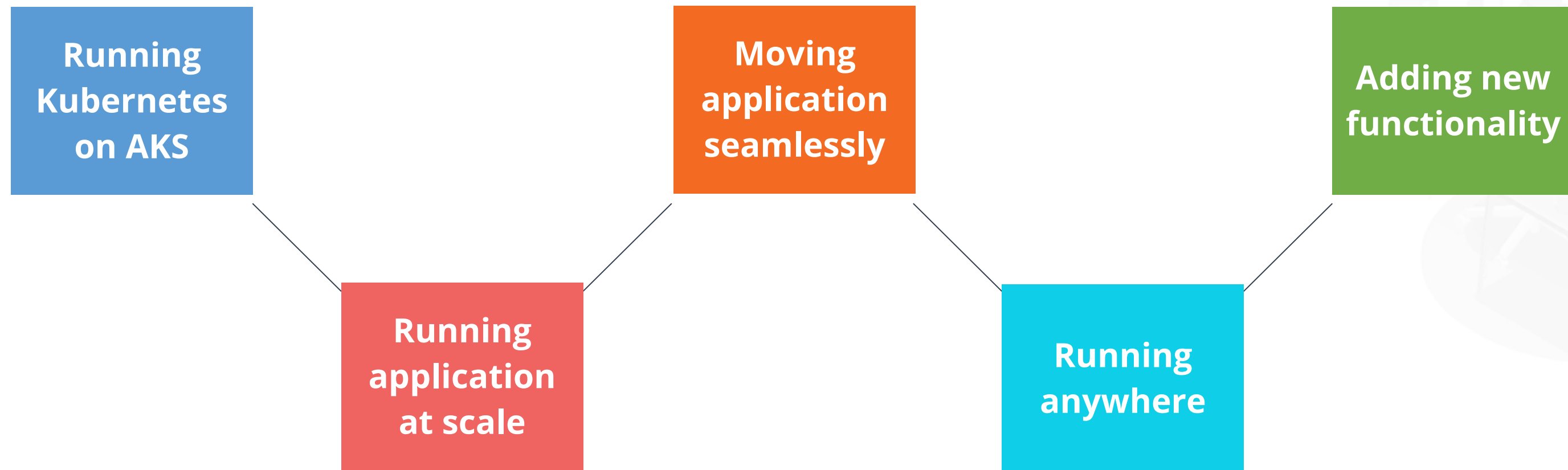
Azure Cloud Platform



- Azure is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications.
- As part of the cloud solution to clients, Microsoft provides special integration of Kubernetes with the Azure Cloud Platform.

Azure Features

Kubernetes is an open-source platform that can be used to run containerized applications anywhere, without needing to change operational tooling. Azure-specific features of Kubernetes include:



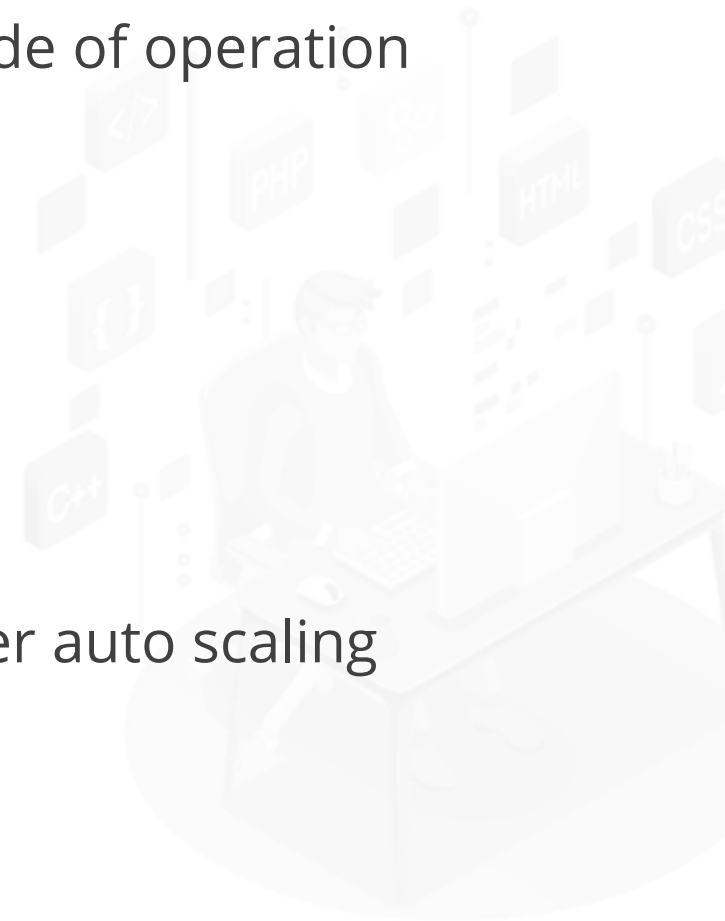
Key Features of Kubernetes on Azure

Enhanced workload and network security

Automated mode of operation

High performance Kubernetes applications

Pod and cluster auto scaling



Azure Kubernetes Service

Azure Kubernetes Service (AKS) manages hosted Kubernetes environment and makes it easy to deploy and manage containerized applications in Azure.

1

The AKS environment is enabled with features such as automated updates, self-healing, and easy scaling.

2

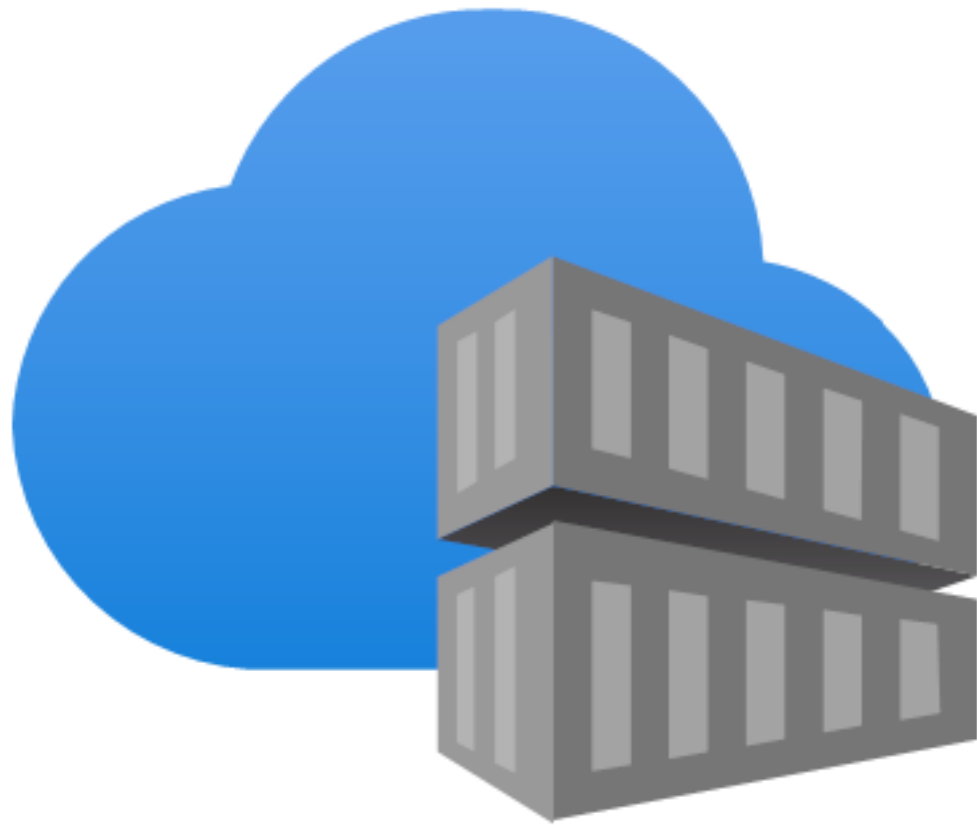
Azure handles vital tasks including health monitoring and maintenance.

3

Azure manages Kubernetes master nodes.

Azure Container Registry

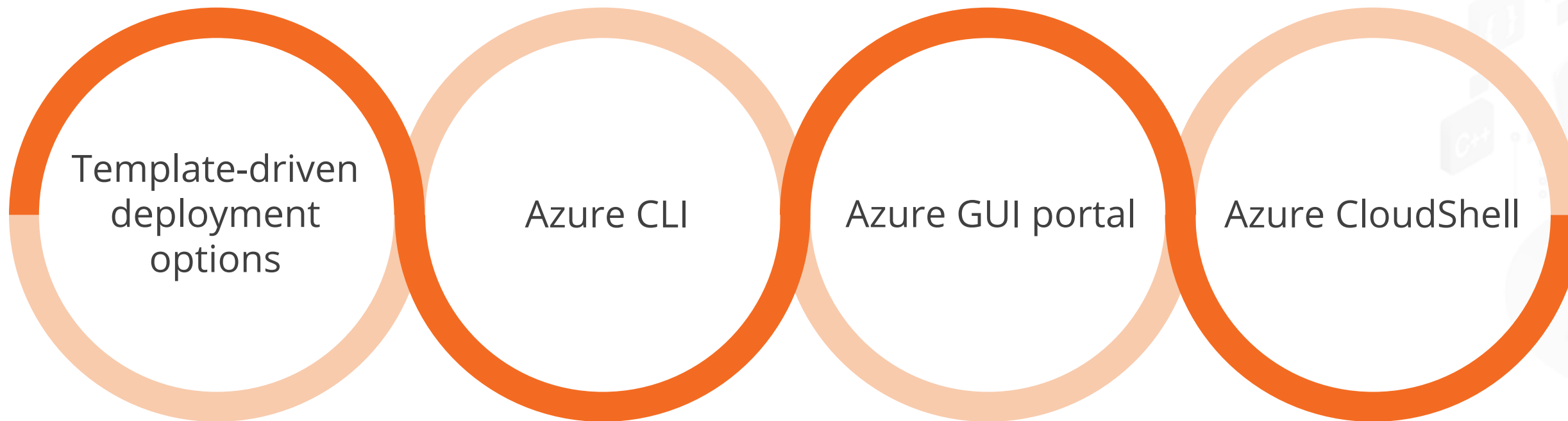
Azure Container Registry is a private registry service that allows users to create, store, and manage container images and related artifacts.



Users can utilize the Azure portal to build an Azure container registry instance and then use Docker commands to push or pull a container image.

AKS Components

AKS provides the following components which help in managing the Kubernetes instance on the Azure instance:



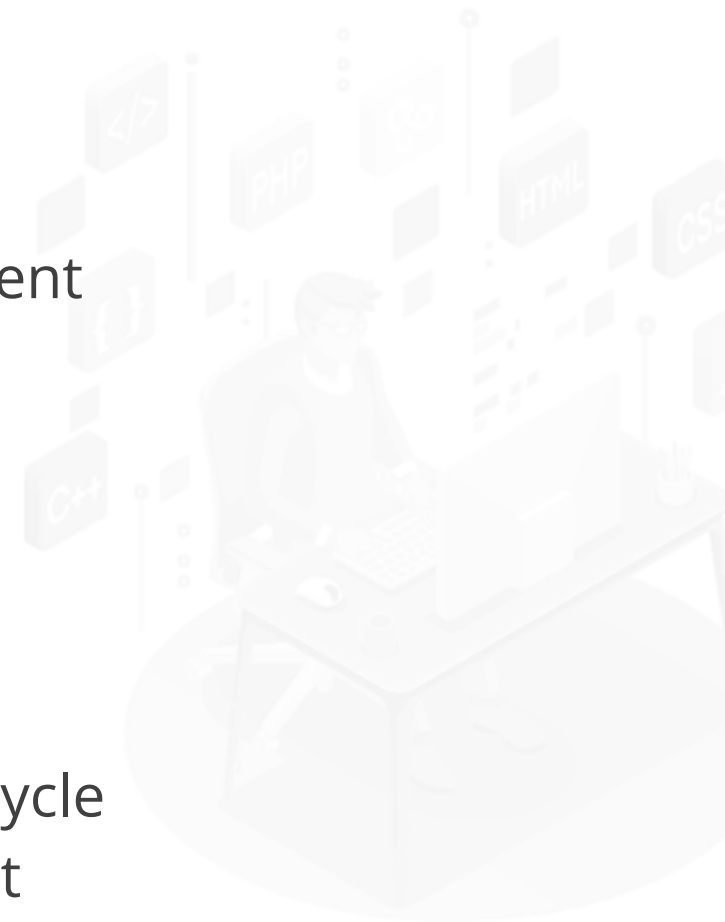
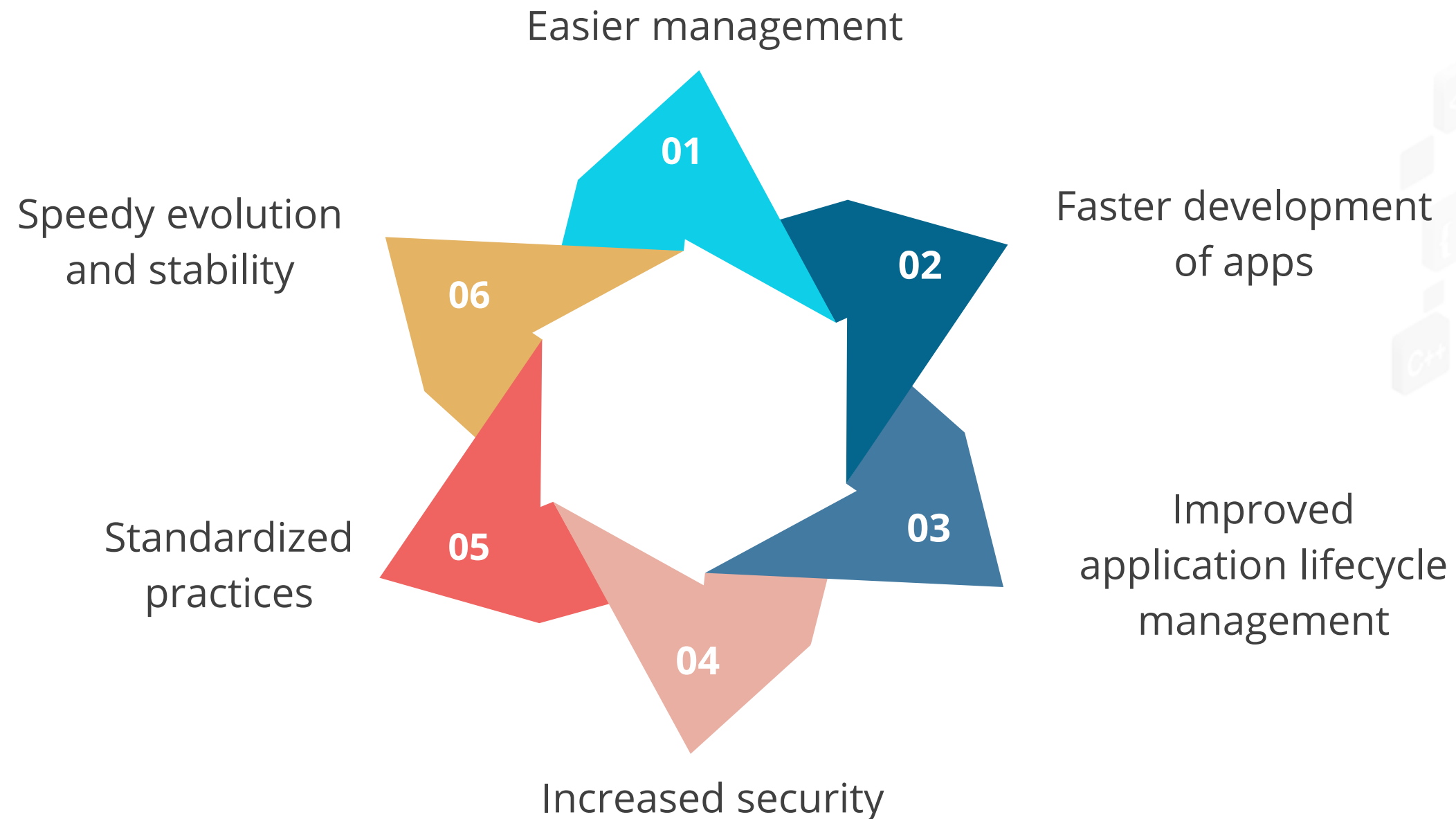
Cloud Interoperability

Cloud interoperability refers to a system's capacity to interact and work efficiently across several cloud platforms.



Benefits of Leveraging AKS

Some of the benefits of leveraging AKS are as follows:



Advantages of AKS over Local Deployment

AKS offers the following advantages over a local deployment:

1

Reduced management overhead for the organization

2

Option for integrating Active Directory services

3

Multiple forms of access to the cluster

4

Automatic configuration of all Kubernetes master nodes

5

Advanced networking features such as HTTP routing

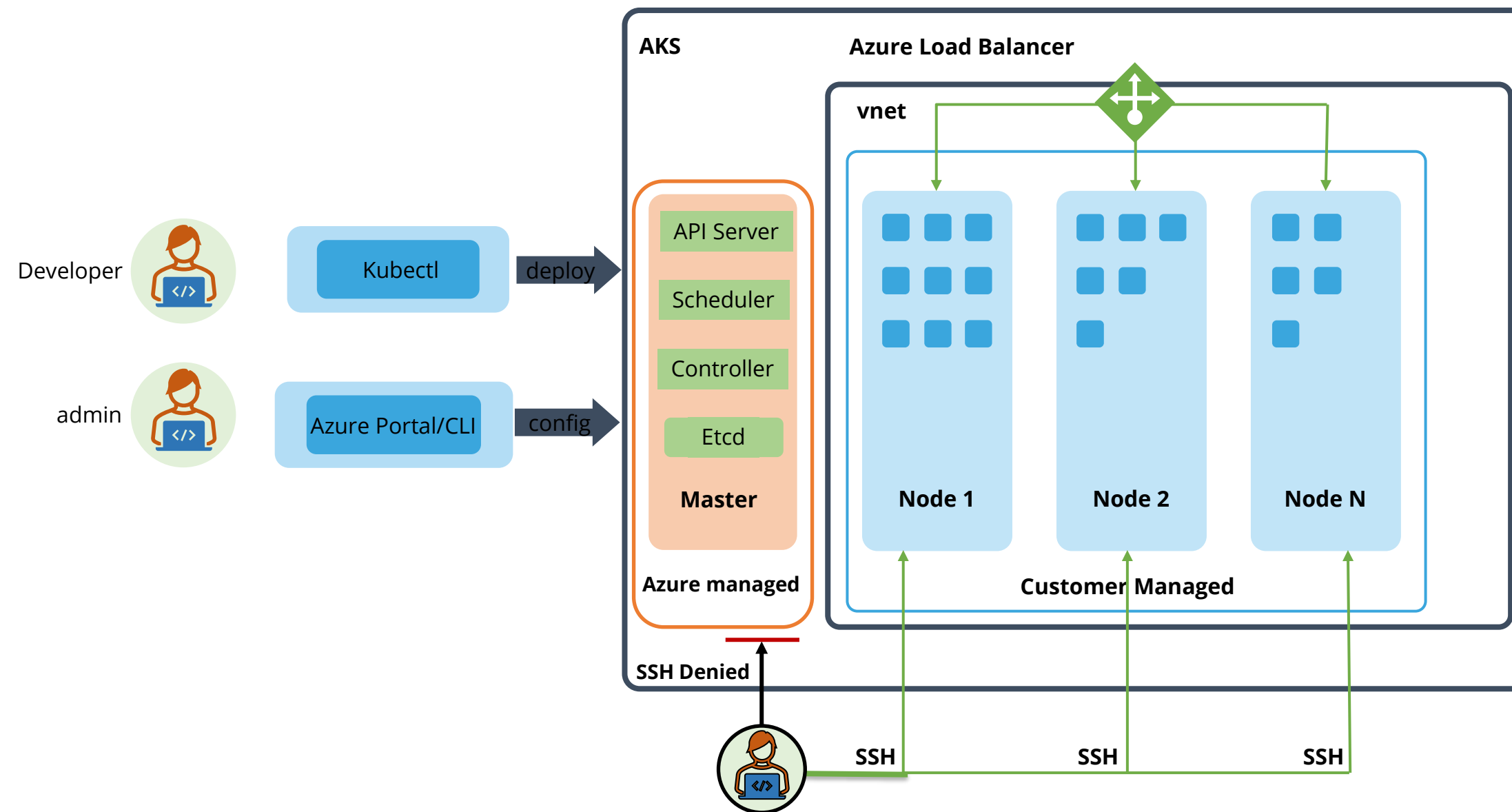
6

Role-based Access Control for increased security

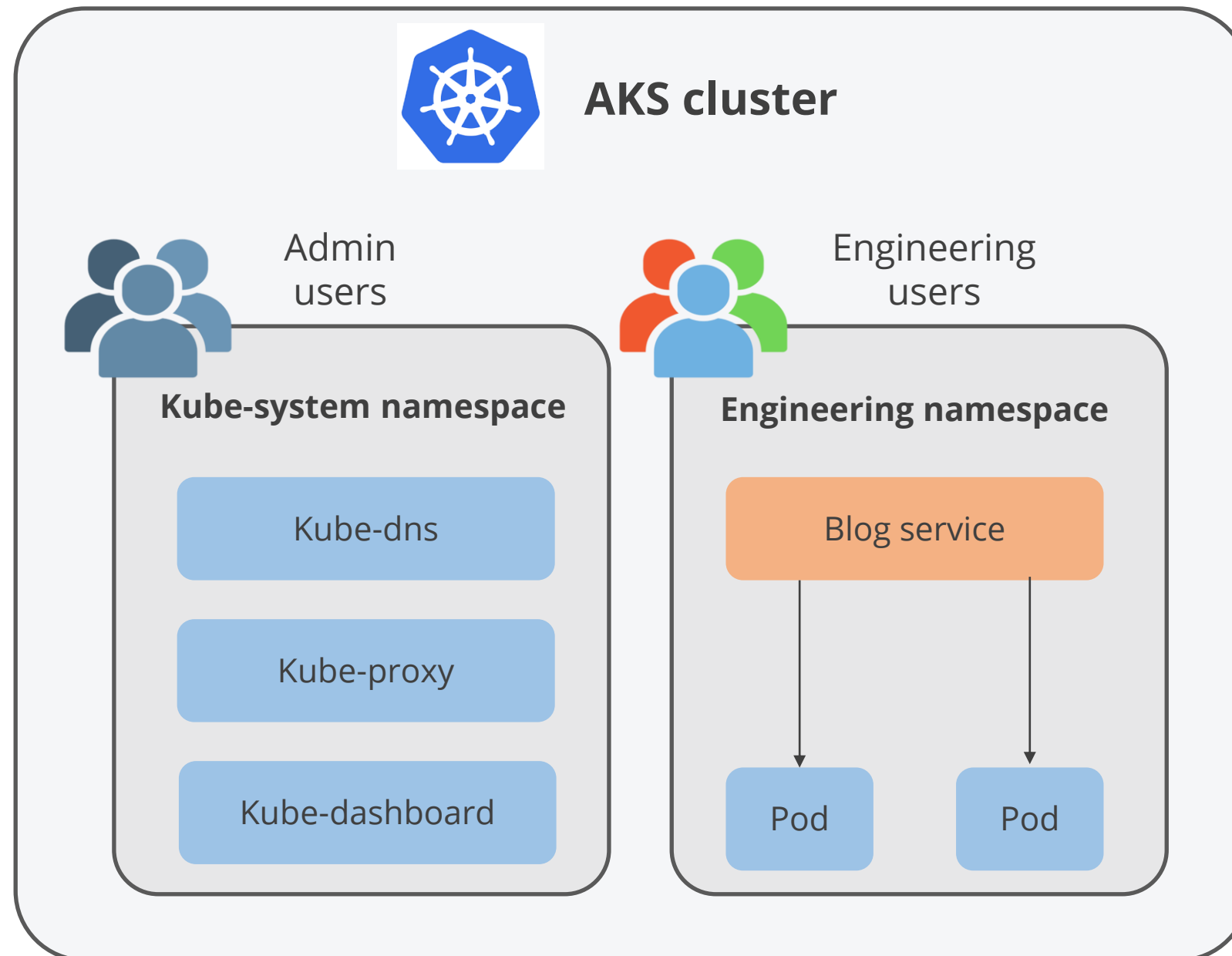
AKS Components

AKS Cluster Creation

Users can either create the cluster in the Azure portal or use the Azure CLI. When users create the cluster, they can use the resource manager templates to automate cluster creation.



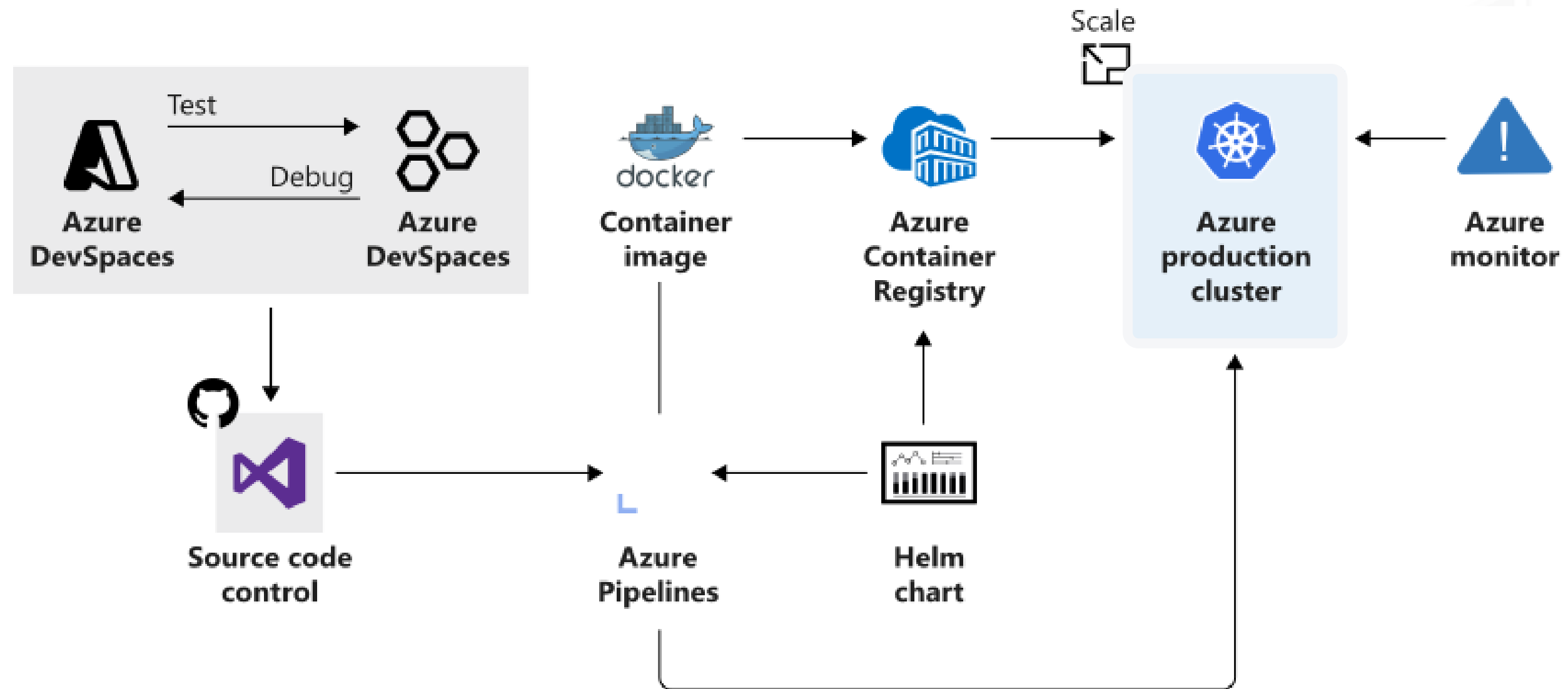
Namespaces in AKS



Kubernetes resources, such as pods and deployments, are logically grouped into a namespace to divide an AKS cluster and restrict create, view, or manage access to resources.

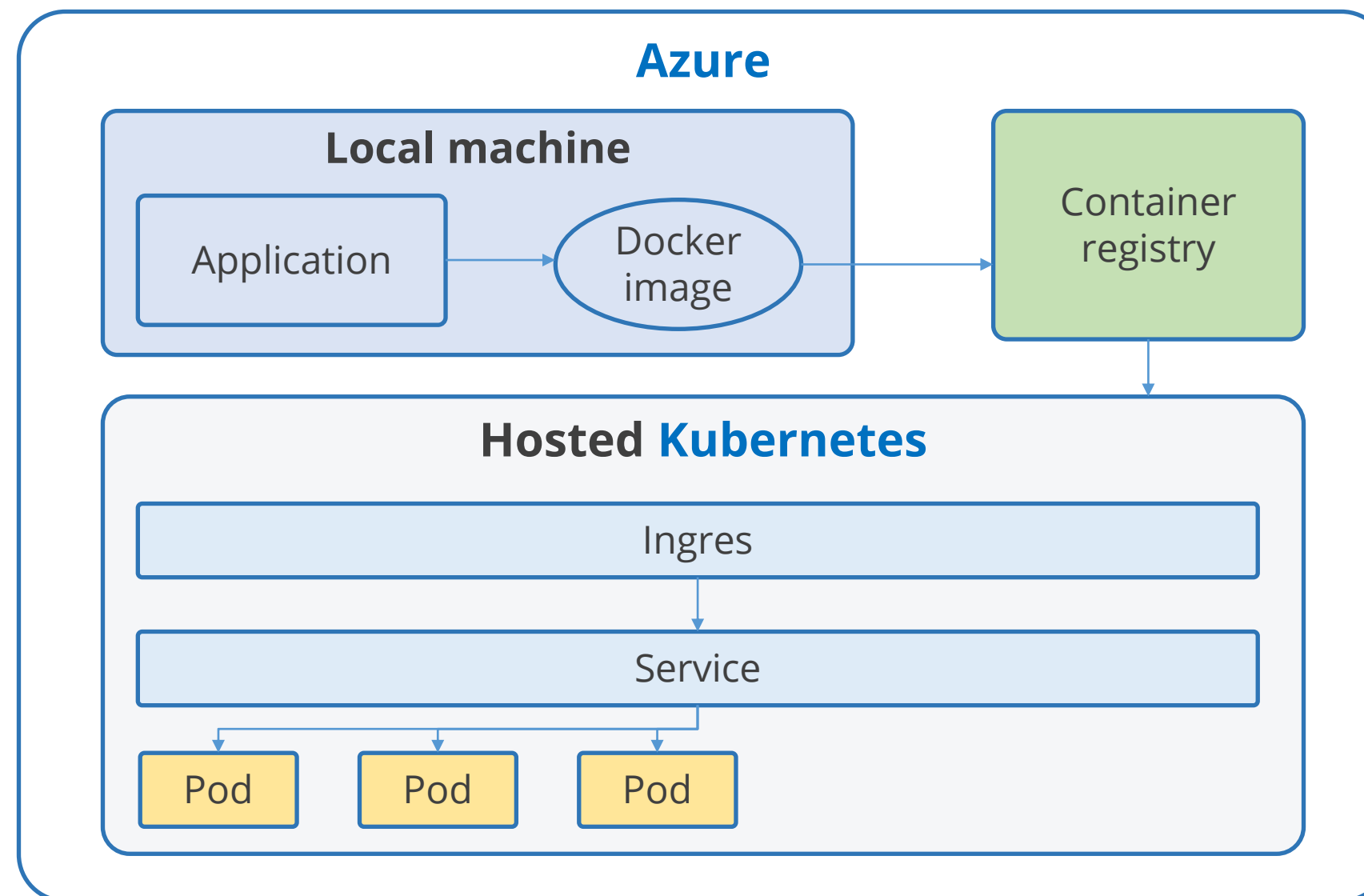
Creating Workloads

AKS supports Docker image format, which means that the users can create a workload in any development environment, package the workload as a container, and deploy the container as a Kubernetes pod.



Creating and Deploying Pods

The Pod creation and deployment in AKS is like the process in the local environment. These Pods can then be controlled remotely using the AKS tools.

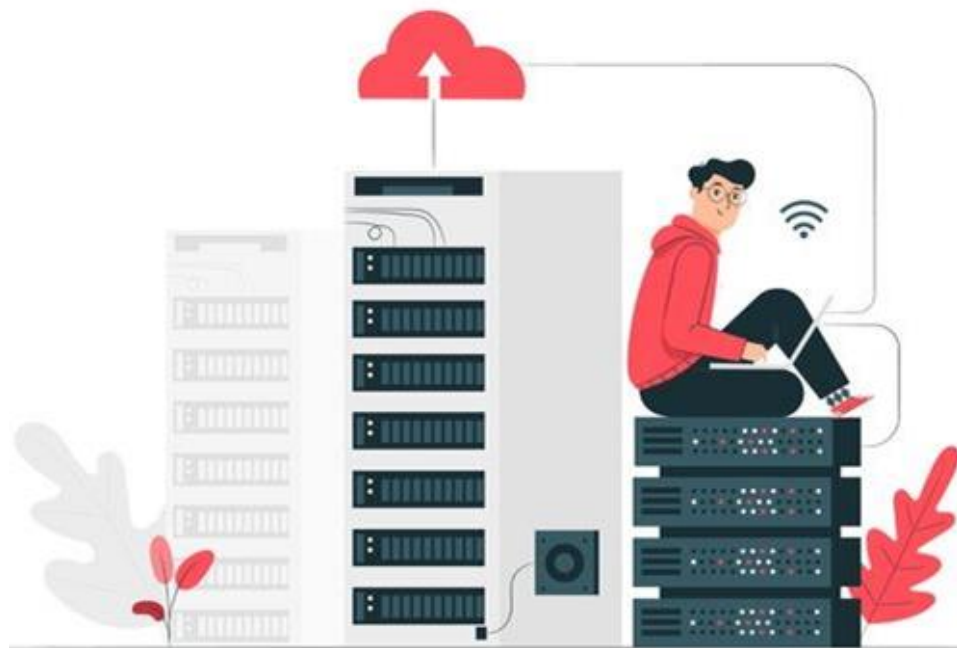


Scheduling Pods



- Kubernetes uses the concepts of **taints** and **tolerances** to schedule the pods and services.
- **Taint** is applied to a node to indicate only specific pods can be scheduled on them.
- **Toleration** is applied to a pod, allowing them to tolerate a node's taint.

Storage in AKS Cluster



- To implement a volume in a Kubernetes cluster, users need to define a **PersistentVolumeClaim** for a specific storage class.
- A **storage class** represents the characteristics of the underlying storage, such as performance or support for shared access.
- **PersistentVolumeClaim** includes information about the required **access mode** and **volume size**.
- Azure has storage volume types called **Azure Disk** and **Azure files**.

Create a Kubernetes Cluster Using AKS



Duration: 15 mins

Problem Statement:

To set up the prerequisites and create a Kubernetes cluster using AKS.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Setting up the prerequisites for configuring an AKS cluster
2. Creating a Kubernetes cluster using AKS service



Create Workloads in an AKS Cluster



Duration: 15 mins

Problem Statement:

To create workloads such as Pods and Deployments in the AKS cluster.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Creating a namespace
2. Creating a Pod
3. Creating a Deployment



Connecting an AKS cluster from the Windows PowerShell



Duration: 25 mins

Problem Statement:

To connect an AKS cluster from Windows PowerShell.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Setting up a storage account resource
2. Creating a new file share for the storage account
3. Setting up Azure Cloud Shell
4. Creating a Deployment and rolling out an update for it



Expose a Deployment as a Service



Duration: 15 mins

Problem Statement:

To expose a Deployment as a Service.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Connecting an Azure cloud shell to the Kubernetes cluster
2. Exposing a Deployment as a Service



Deploy Application with Load Balancer



Duration: 15 mins

Problem Statement:

To deploy the application with a load balancer using AKS bash.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Deploying the application with a load balancer using AKS bash



Dynamic Volume



Duration: 15 mins

Problem Statement:

To create a dynamic and a persistent volume with Azure disks in Azure Kubernetes Service (AKS).

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Creating a persistent volume claim using AKS bash



Monitoring Clusters and Node Logs



Duration: 10 mins

Problem Statement:

You've been assigned a task to monitor cluster health using Azure Monitor.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Monitoring cluster health using Azure Monitor
2. Checking logs of a running Pod using Azure Log Analytics
3. Checking logs of a running Pod using Azure Cloud Shell



Kubernetes Cluster Setup In PowerShell Using Azure CLI



Duration: 10 mins

Problem Statement:

You've been assigned a task to set up a Kubernetes Cluster inside the Windows PowerShell using Azure CLI.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Installing Azure CLI
2. Setting up the Azure credentials
3. Installing a Kubernetes cluster



Update an Application



Duration: 10 mins

Problem Statement:

You've been assigned a task to deploy new versions of an application.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Creating a Deployment using the image `portainer/portainer:1.24.2-alpine`
2. Verifying the Deployment and Pods
3. Scaling the Deployment for a better understanding
4. Setting a new image version for the portainer Deployment
5. Verifying the Pods



Upgrade an AKS Cluster



Duration: 10 mins

Problem Statement:

You've been asked to upgrade the k8s-Cluster version 1.22.6 to 1.23.3.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Navigate to the k8s-Cluster configuration
2. Click on the Kubernetes version 1.22.6
3. Click on the upgrade version
4. Scroll and select change Kubernetes version from 1.22.6 to 1.23.3
5. Upgrade the Kubernetes version to 1.23.3
6. Verify the version



Install NGINX Ingress Controller Using Helm



Duration: 5 mins

Problem Statement:

You have been asked to install the NGINX Ingress Controller on a Kubernetes cluster using Helm.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Verifying the Helm version
2. Installing the NGINX Ingress Controller using Helm
3. Validating the NGINX Ingress Controller



Key Takeaways

- KaaS assists users in migrating to cloud-native enabled Kubernetes-based platforms and managing the lifetime of K8s clusters.
- Azure Kubernetes Service (AKS) manages the hosted Kubernetes environment and makes it simple to deploy and manage containerized applications in Azure.
- Clusters can be created in the Azure portal or the Azure CLI. When a cluster is created, the Resource Manager uses templates to automate cluster creation.
- Kubernetes resources, such as pods and deployments, are logically grouped into a namespace to divide an AKS cluster and restrict create, view, or manage access to resources.



Deploy MySQL Using Azure Dynamic Storage Class

Duration: 15 Min



Project agenda: To deploy MySQL using Azure dynamic storage class

Description:

you have been asked to design a solution to deploy MySQL using Azure dynamic storage class. You need to check the storage classes provided by the Azure Kubernetes Service (AKS) for deploying the MySQL Pods with Azure disk as Persistent Volume.

Perform the following:

1. Creating an AKS cluster
2. Creating an Azure disk for dynamic persistent volume provisioning
3. Deploying MySQL Pods with Azure disk as PV