

**Istio with AKS (Azure Kubernetes Service)**

**Course Number:** CLD-130
**Duration:** 2 days

**Overview**

This hands-on course teaches learners to implement and manage Istio on Azure Kubernetes Service (AKS). Through practical exercises, learners gain proficiency in leveraging Istio for traffic control, enhancing security, achieving comprehensive observability, and streamlining service-to-service communication within microservices deployments.

**Prerequisites**

All participants must have taken [Introduction to Docker and Kubernetes](file:////training/docker-kubernetes-introduction) or have equivalent experience.

**Materials**

All Istio with AKS training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

Attendees will not need to install any software on their computers for this class. The class will be conducted in a remote environment that Accelebrate will provide; students will only need a local computer with a web browser and a stable Internet connection. Any recent version of Microsoft Edge, Mozilla Firefox, or Google Chrome will work well.

**Objectives**

* Understand the fundamentals of service mesh and Istio
* Deploy Istio on AKS and configure its components
* Manage traffic routing and observe advanced networking in Istio
* Implement Istio security features like mutual TLS and authorization policies
* Gain insights into microservices performance using Istio observability tools

**Outline**

* Introduction
* What is a Service Mesh?
	+ Challenges in microservices networking
	+ Service mesh solutions and benefits
* Istio Overview
	+ Key features: traffic management, security, and observability
	+ Istio architecture: Envoy proxy, control plane components
	+ Sidecar vs. Ambient mode
* Azure Kubernetes Service (AKS) Overview
	+ Benefits of using AKS for container orchestration
	+ Prerequisites for Istio deployment on AKS
* Installing Istio on AKS
	+ Configuring AKS clusters
	+ Installing Istio using Helm or Istioctl
	+ Verifying the installation
* Core Traffic Features
	+ Traffic splitting, retries, and timeouts
	+ Implementing canary deployments and blue-green deployments
* Istio Gateway and VirtualService Resources
	+ Configuring ingress and egress traffic
	+ Path-based and header-based routing
* Monitoring and Telemetry
	+ Istio metrics and logs using Prometheus and Grafana
	+ Distributed tracing with Jaeger
	+ Service graph visualization using Kiali
* Securing Service-to-Service Communication
	+ Mutual TLS (mTLS) setup
	+ Istio authentication policies
* Authorization Policies
	+ Configuring role-based access control (RBAC)
	+ Enforcing security policies
* Optimizing Istio for Performance
	+ Fine-tuning Envoy proxies
	+ Resource management for Istio components
* Scaling and Upgrading Istio
	+ Rolling updates for Istio control plane
	+ Strategies for multi-cluster service mesh
* Troubleshooting Common Issues
* Conclusion