

**Introduction to Karpenter with EKS**

**Course Number:** CLD-126
**Duration:** 3 days

**Overview**

This Karpenter with EKS course introduces Karpenter, an open-source Kubernetes cluster autoscaler. Participants learn how Karpenter works, compare it to other cluster autoscaling mechanisms, and deploy and configure it to optimize Kubernetes workloads.

**Prerequisites**

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

**Materials**

All Karpenter with EKS 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 role of Karpenter in Kubernetes cluster management
* Deploy Karpenter in a Kubernetes cluster
* Configure Karpenter to manage and optimize node provisioning based on workload demands
* Use Karpenter with real-world workloads to scale infrastructure efficiently
* Diagnosing and troubleshooting common issues when using Karpenter

**Outline**

* Introduction
* What is Karpenter?
	+ Key features
	+ Node provisioning
	+ Cost optimization and efficient resource usage
* How Karpenter Works
	+ Scheduling and provisioning workflows
	+ Integration with Kubernetes and cloud providers
* Comparison with Cluster Autoscaler
* Prerequisites for Karpenter
	+ Kubernetes version requirements
	+ Supported cloud environments (AWS focus)
* Creating and Managing Provisioners
	+ Spot instance configurations
	+ Scheduling constraints
* Scaling Applications with Karpenter
	+ Deploying sample applications to trigger scaling
	+ Understanding logs and scaling behavior
* Monitoring and Logging with Karpenter
	+ Metrics and monitoring tools integration
* Troubleshooting
	+ Addressing node scheduling issues
* Threat Model
	+ Threats and mitigations
* Conclusion