Part of **Accenture**

Introduction to Chaos Mesh

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy

Get Your Free Copy Now

Course Number: WA3633

Duration: 1 days

Overview

Course Description

This Chaos Mesh training course introduces learners to the principles of Chaos Engineering using Chaos Mesh, an open-source chaos testing platform for Kubernetes. Attendees gain practical skills to inject, monitor, and mitigate failures in distributed systems, ensuring system resilience and reliability.

Skills Gained

- Recognize the role of controlled failure injection in system resilience
- Learn the architecture and components of Chaos Mesh
- Install and configure Chaos Mesh in a Kubernetes environment
- Create and execute Chaos Experiments for common failure scenarios
- Analyze the impact of chaos experiments on system performance
- Use monitoring tools to observe system behavior during experiments
- Evaluate metrics and logs to identify potential system improvements
- Develop strategies to enhance resilience based on chaos experiments

- Incorporate Chaos Engineering into CI/CD pipelines for proactive failure testing

Prerequisites

All learners should have taken Intro to Docker or Kubernetes or have equivalent knowledge.

Audience

Course Details

Introduction to Chaos Engineering

- What is Chaos Engineering?
- Goals and benefits of controlled failure testing
- Chaos Engineering vs. traditional testing

Introduction to Chaos Mesh

- Overview of Chaos Mesh and its architecture.
- Installing Chaos Mesh in a Kubernetes environment.

Running Chaos Experiments

- Understanding Chaos Experiments and fault injection
- Pod failures
- Network delays or disconnections
- CPU and memory stress tests
- DNS disruptions

Monitoring and Analyzing Results

- Observing system behavior using tools like Prometheus and Grafana
- Analyzing experiment results with Chaos Dashboard

Improving Resilience and Automating Chaos

- Strategies for addressing discovered weaknesses
- Incorporating Chaos Mesh into CI/CD pipelines

Conclusion