SZKOLENIE ŚREDNIO ZAAWANSOWANE

Istio w praktyce

ISTIO

Czas trwania: 2 dni (16h)

Zasady, koncepcje i praktyczne rozwiązania dotyczące wdrażania mikroserwisów

Cele szkolenia

- Solidne oraz kompleksowe wprowadzenie do Istio service mesh
- Praktyczne poznanie możliwości Istio w kontekście realizacji zadań takich jak: zarządzanie ruchem, tracing, zapewnianie bezpieczeństwa
- Poznanie Istio w stopniu umożliwiającym jego wdrożenie i zastosowanie w realnym projekcie

Zalety

- Rozbudowana część warsztatowa obejmująca między innymi konfigurację i zarządzanie klastrem
- Wzorce, dobre praktyki, a także sposoby rozwiązywania typowych problemów spotykanych podczas wdrażania Istio
- Praktyka przed teorią - wszystkie szkolenia technologiczne prowadzone są w formie warsztatowej. Konieczna teoria jest wyjaśniana na przykładzie praktycznych zadań
- Konkretne umiejętności - w ramach każdego szkolenia rozwijamy praktyczne umiejętności związane z daną technologią i tematyką
- Nauka z praktykami - wszyscy trenerzy na co dzień pracują w projektach, gwarantuje to dostęp do eksperckiej wiedzy i praktycznego know-how

Dla kogo?

- Devopsi, administratorzy i architekci chcący projektować, tworzyć oraz utrzymywać rozwiązania oparte o Kubernetes i Istio

Wymagania

- Praktyczna znajomość rozwiązań kontenerowych (Docker) oraz Kubernetes

Program

1. Wprowadzenie
    a. Istio jako przykład service mesh
    b. Architektura i najważniejsze funkcjonalności Istio
    c. Istio control plane components
    d. Istio data plane components

    e. Instalacja i modele wdrożeniowe
2. Service proxy (Envoy)
    a. Czym jest i jak działa service proxy?
    b. Envoy jako implementacja service proxy w Istio
    c. Sidecar injection
    d. Konfiguracja
3. Bezpieczeństwo
    a. Bezpieczna komunikacja
    b. Uwierzytelnianie (mTLS, end-user authentication)
    c. Autoryzacja dostępu (protokoły http i tcp, ingress, wykorzystanie tokenów JWT)
    d. Konfigurowanie i zarządzanie polisami
    e. Zarządzanie tożsamością (SPIFFE)
    f. Workload API
4. Zarządzanie ruchem
    a. Networking API (ServiceEntry, DestinationRule, VirtualService, Gateway)
    b. Routing i kontrola ruchu w klastrze
    c. Ingress I egress
    d. Load-balancing
    e. Wysoka dostępność (wykrywanie i reagowanie na potencjalne problemy, równoważenie obciążenia)
5. Observability
    a. Adaptery - zbieranie i wykorzystanie metryk
    b. Wizualizacja klastra/service mesh
    c. Praca z logami
    d. Distributed tracing
6. Elementy zaawansowane
    a. Debugging oraz troubleshooting
    b. Planowanie topologii klastra
    c. Wydajność i skalowanie
    d. Najlepsze praktyki
    e. Integracja