

Tên đề án: Triển khai OpenShift

Mục đích:

Tiến hành triển khai hệ thống OpenShift OKD 3.11 và tích hợp triển khai một ứng dụng mẫu theo mô hình kiến trúc microservices nhằm kiểm chứng khả năng vận hành và triển khai ứng dụng trên nền tảng này.

Giới thiệu:

OpenShift OKD 3.11 là phiên bản mã nguồn mở (community edition) của Red Hat OpenShift Container Platform, dựa trên Kubernetes, được quản lý và phát triển bởi cộng đồng.

Với OKD, ta có thể trải nghiệm OpenShift miễn phí, phù hợp cho môi trường phát triển , thử nghiệm, nghiên cứu hoặc doanh nghiệp nhỏ không cần hỗ trợ thương mại từ Red Hat

Điểm đặc biệt của OpenShift OKD 3.11 là phát triển dựa trên:

- + Kubernetes 1.11
- + Docker (container engine)
- + Hệ thống Origin Orchestration
- + Sử dụng hệ thống Ansible để cài đặt và quản lý cụm

Các tính năng nổi bật của OpenShif OKD 3.11

Tính năng	Mô tả
Web console	Giao diện quản lý đồ họa
Persistent storage support	Hỗ trợ nhiều hệ thống lưu trữ như NFS, GlusterFS
RBAC	Quản lý người dùng và quyền truy cập
Integrated CI/CD pipelines	Có hỗ trợ Jenkins tích hợp
Template & Helm	Hỗ trợ triển khai ứng dụng bằng template YAML

Cấu trúc hành phần OKD 3.11 của project:

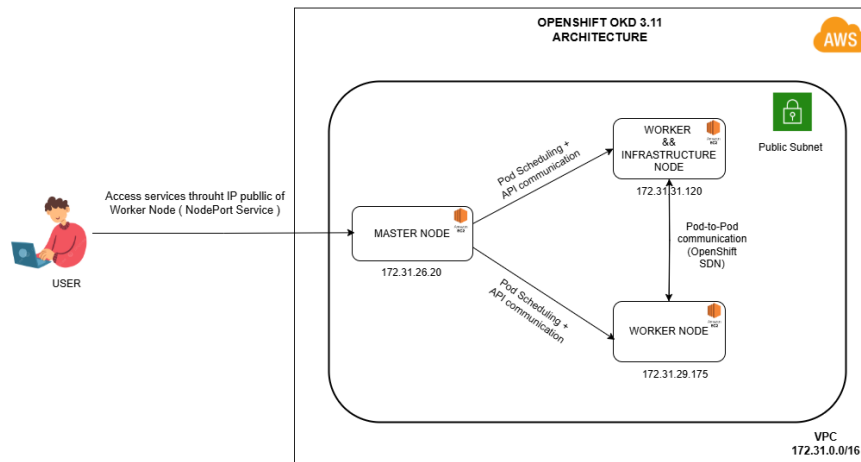
Master nodes: Chạy API server, scheduler, controller

Infrastructure nodes: Chạy router, registry, metrics, logging

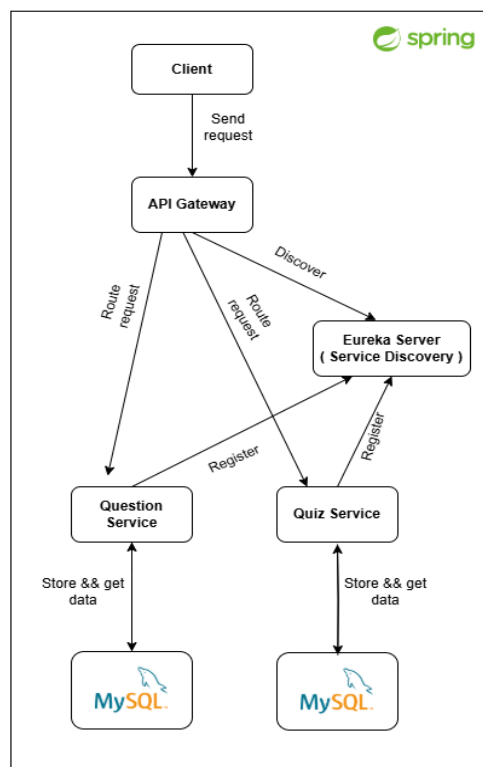
Worker nodes: Chạy các ứng dụng (pods)

Thực hiện triển khai trên 3 máy ảo EC2 trên AWS. Trong đó có 1 node đảm nhiệm 2 vai trò Infrastructure and Worker node.

System architecture of OKD 3.11:



Microservice Architecture:



Mô tả chi tiết:

+ Sử dụng Ansible để tự động hóa quá trình cài đặt và cấu hình OpenShift OKD 3.11 trên nền tảng hạ tầng AWS.

+ Cụm được triển khai bao gồm:

- **1 Master Node:** chịu trách nhiệm quản lý cluster, xử lý các yêu cầu API, điều phối Pods, và lưu trữ trạng thái hệ thống thông qua etcd.
- **1 Worker Node:** nơi triển khai các Pod ứng dụng.
- **1 Worker kiêm vai trò Infrastructure Node:** đảm nhiệm cả vai trò chạy ứng dụng và cung cấp các dịch vụ hệ thống như routing, registry, monitoring...

+ Triển khai ứng dụng Spring Boot theo kiến trúc Microservice

Ứng dụng được thiết kế theo mô hình **microservices**, bao gồm:

- **Eureka Server:** đóng vai trò làm *service discovery*, nơi các service đăng ký và tìm kiếm lẫn nhau.
- **API Gateway:** xử lý toàn bộ request từ phía client và định tuyến tới các service tương ứng.
- **Question Service** và **Quiz Service:** là 2 service chính đại diện cho các nghiệp vụ của hệ thống.

Mỗi service được:

- Đóng gói thành **Docker image**.
- Triển khai lên OKD bằng các file yaml deployment , service ,...
- Đăng ký với Eureka Server tự động.
- API Gateway kết nối đến các service thông qua service discovery

+ Kiểm thử hoạt động hệ thống bằng Postman

Sử dụng công cụ **Postman** để gửi các request HTTP đến API Gateway.

Truy cập các service thông qua NodePort Service (Expose các port trên các Worker Node và truy cập thông IP public của Work Node)

Một số request kiểm thử:

- GET /question/all → kiểm tra truy xuất danh sách câu hỏi.

- POST /quiz/create → tạo quiz mới.

Kết quả kỳ vọng:

- Cụm OpenShift hoạt động ổn định, các node giao tiếp tốt
- Các service microservices triển khai thành công và có thể mở rộng độc lập
- API Gateway hoạt động đúng chức năng định tuyến
- Các API có thể gọi và phản hồi đúng qua Postman, chứng minh hệ thống hoạt động hoàn chỉnh