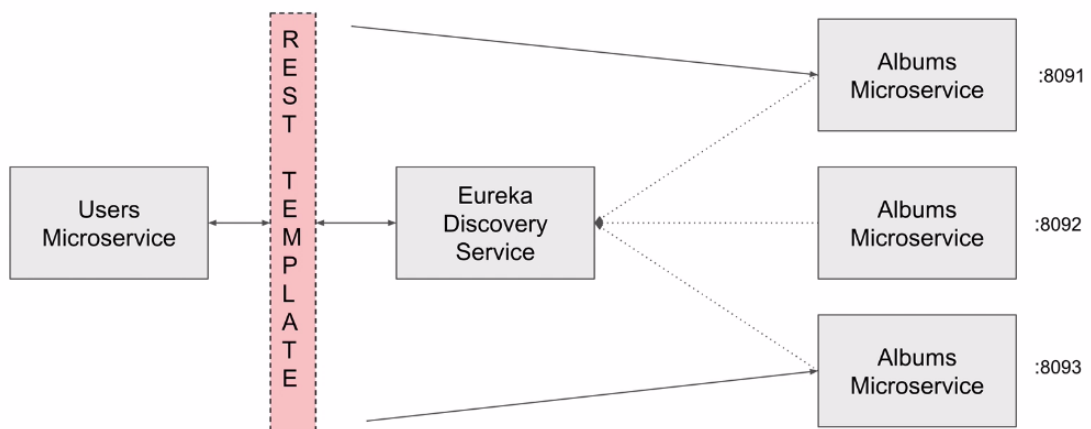


Photo Album App Microservices

1 Overall Architecture

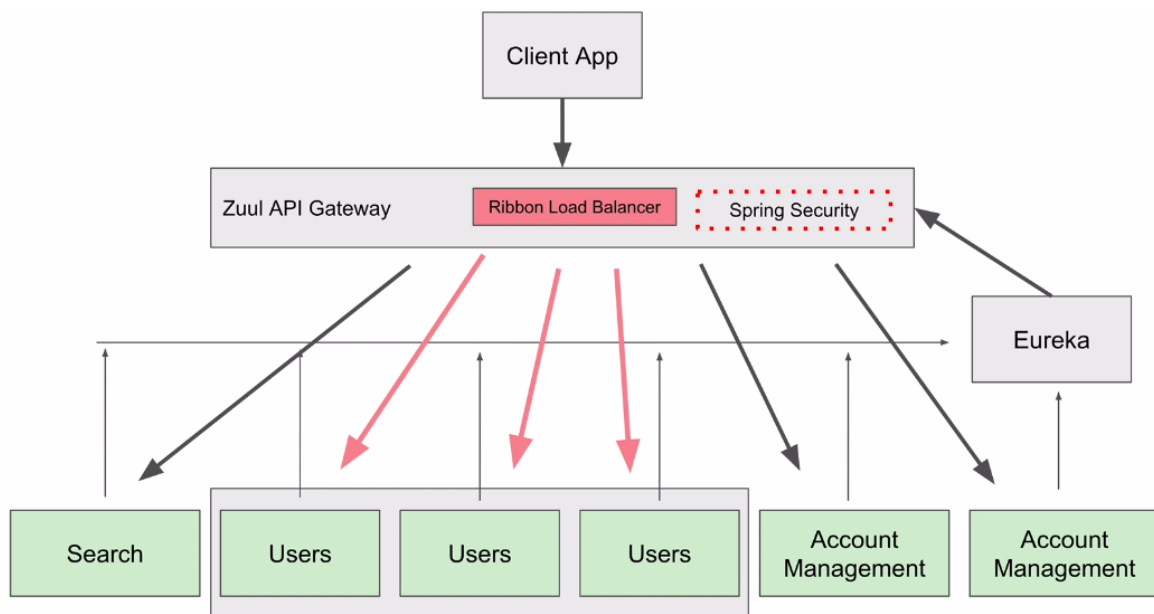
- RESTful Microservices (User Services + Album Services)
- Eureka Discovery Service
- Zuul API Gateway (with Load Balancer and Spring Security)
- Spring Cloud Config Server



`http://albums-ws/albums/<album-id>`

albums-ws

2 API gateway



3 User Services

User services APIs:

1) create

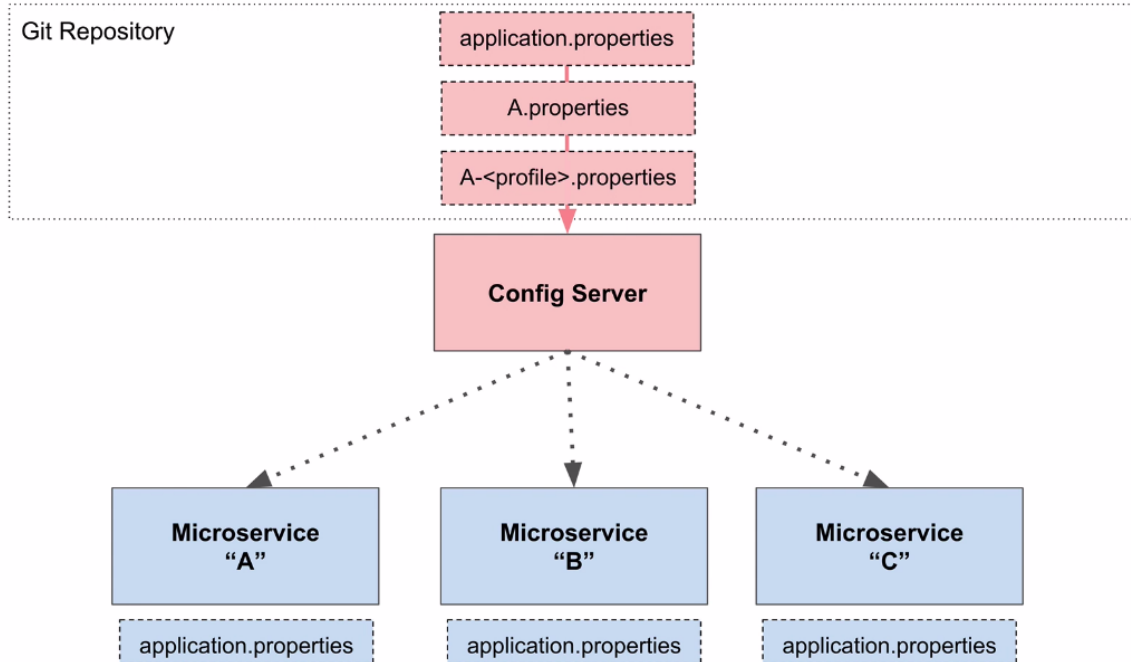
2) get

3) login (authentication): JSON Web Token (JWT)

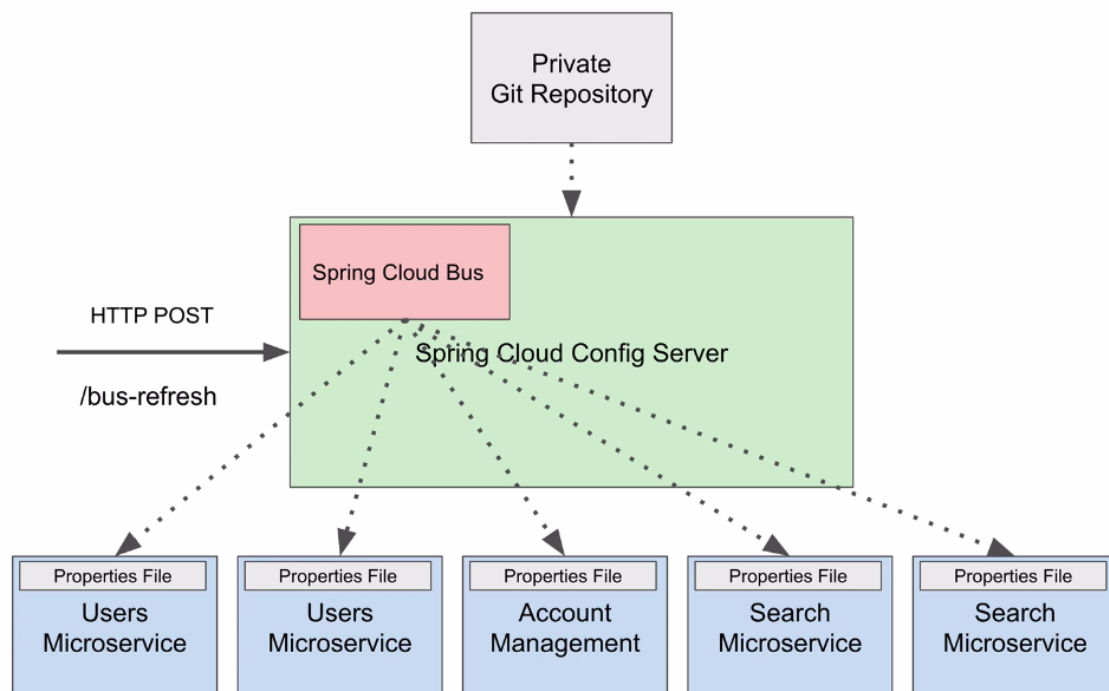
authorization (register)

4 Config Server

1) config file hierarchy



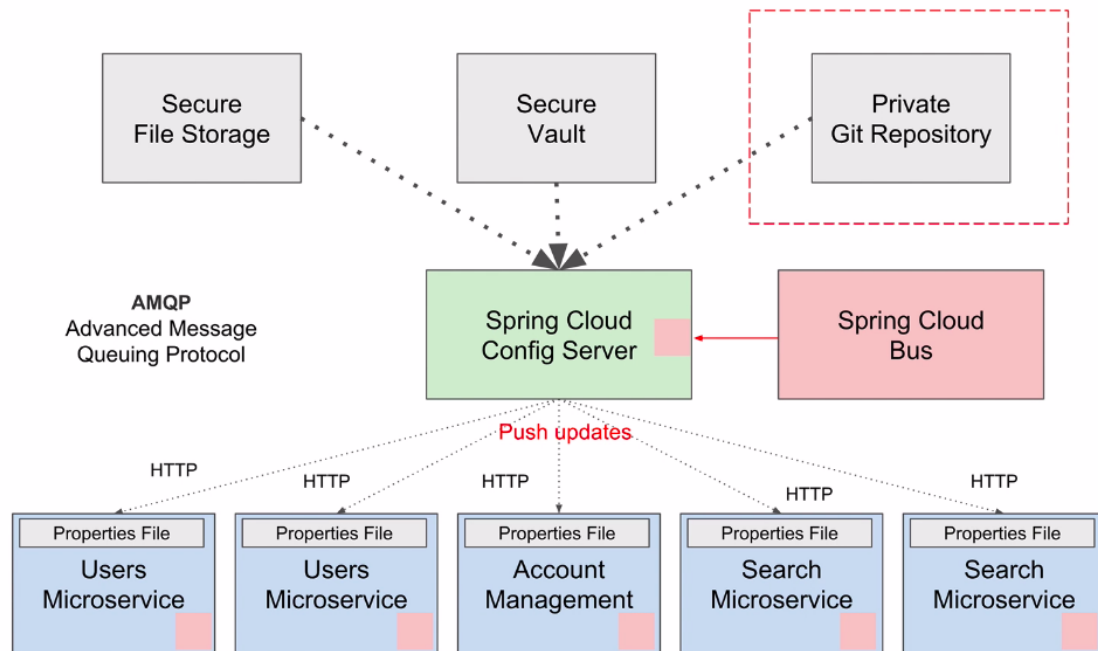
2) Static Config



3) Dynamic Config

Spring Cloud Bus

Rabbit MQ



5 Data Encryption

Java Cryptography Extension

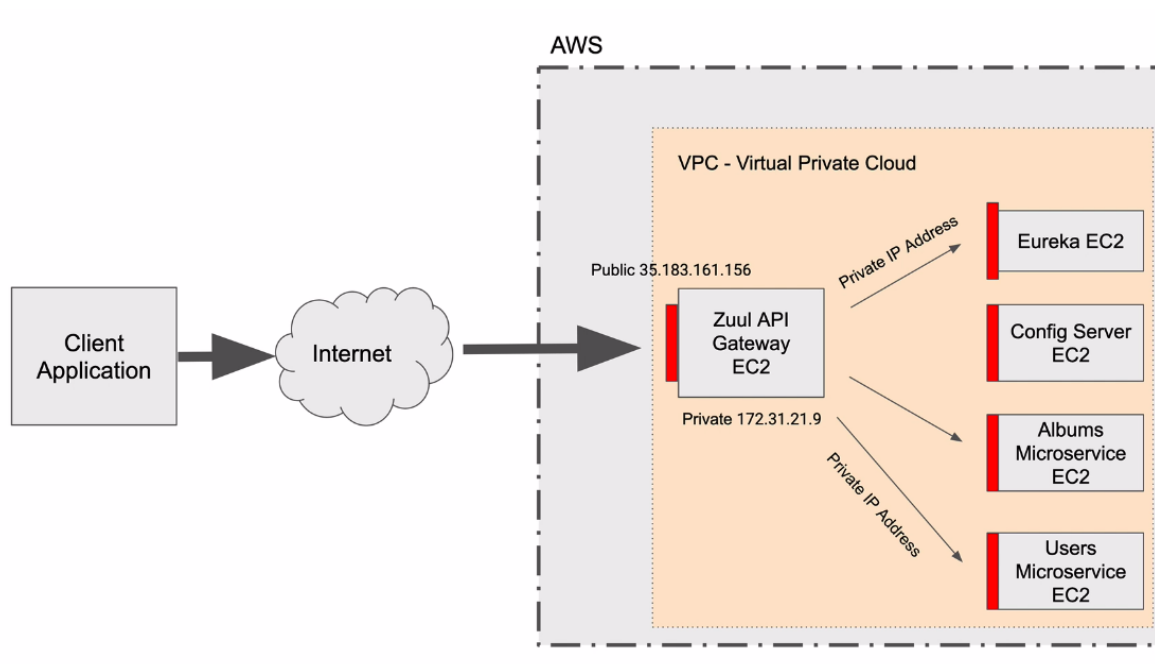
6 Tracing and Logging

- 1) Distributed Tracing of HTTP requests: Spring Cloud Sleuth and Zipkin
- 2) Logging: aggregate log files in one place(Centralized logging) with ELK stack (Elasticsearch, Logstash, Kibana)

7 Deployment

Docker + AWS EC2 instances

1) Architecture



2) Plan and Sequence

