Bipin Kumar

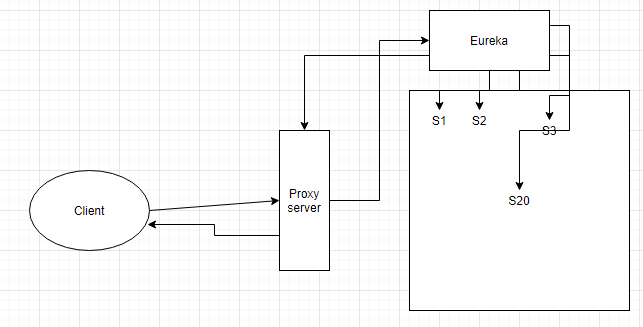
Shael.dhn88@gmail.com

Zuul

Zuul Setup

Suppose we have 20 services like S1, S2 … S20 all register to Eureka. So, if internal call happens between these services then it will go to eureka and found the service registry and call those service.

Now suppose if any external client wants to use one of your service can you give it all the registry present in Eureka. So instead of giving all the registry present in Eureka we will provide a proxy server. So now client will call proxy server and proxy server will go to eureka get the registry and based on service call return data. Zuul is one of the libraries which is used as API gateway. It also implements hystrix. Suppose I want to secure my all service we will implement at zuul, similarly we can also configure process like pre-processing or post processing of any request. Also, we can write routing logic also if client wants to use only some of the services.



1. Add below dependency in zuul project pom.xml.

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-zuul</artifactId>

</dependency>

1. In main class add below annotation.

@EnableZuulProxy

@EnableDiscoveryClient

1. Add below property in yml file.

spring:

application:

name: gateway

server:

port: 1234

error:

whitelabel:

enabled: false

hystrix:

command:

default:

execution:

timeout:

enabled: false

eureka:

instance:

instance-id: ${spring.application.name}:${random.value}

client:

serviceUrl:

defaultZone: http://localhost:5001/eureka

logging:

pattern:

console: "%clr(%d{yyyy-MM-dd HH:mm:ss.SSS}){faint} %clr(%5p) %clr(${PID:- }){magenta} %clr(---){faint} %clr([trace=%X{X-Trace-Id:-},span=%X{X-Span-Id:-}]){yellow} %clr([%15.15t]){faint} %clr(%-40.40logger{39}){cyan} %clr(:){faint} %m%replace(%xException){'\n','\u2028'}%nopex%n"

if we open postman and make service call using zuul. We have to do below thing. Here we use convention to call the rest api as you can see in below url.

<http://localhost:1234/booking-service/cargobooking>

when you hit above url with proper data you can see success response. Now let’s add zuul related property. In below we can configure the resource url with /api. We also don’t want external client to know real service name so we can configure alias for that using below routes rule. Similarly, we want to not call some service then we can also configured that using ignored-services.

zuul:

prefix: /api

ignored-services: /\*

routes:

booking-service: /bs/\*

routing-service: /rs/\*

Suppose if we want to configured some pre or post logic to be executed before call we can use zuulfilter

If method shouldFilter return true then only logic inside run method will be executed.

filterType: it will determine when to execute filter pre or post.

filterOrder: Suppose we have more than one filter then based on the value you configured this filter will be executed. Less the value more the priority.