

fox9916的博客

冰与火

RSS订阅

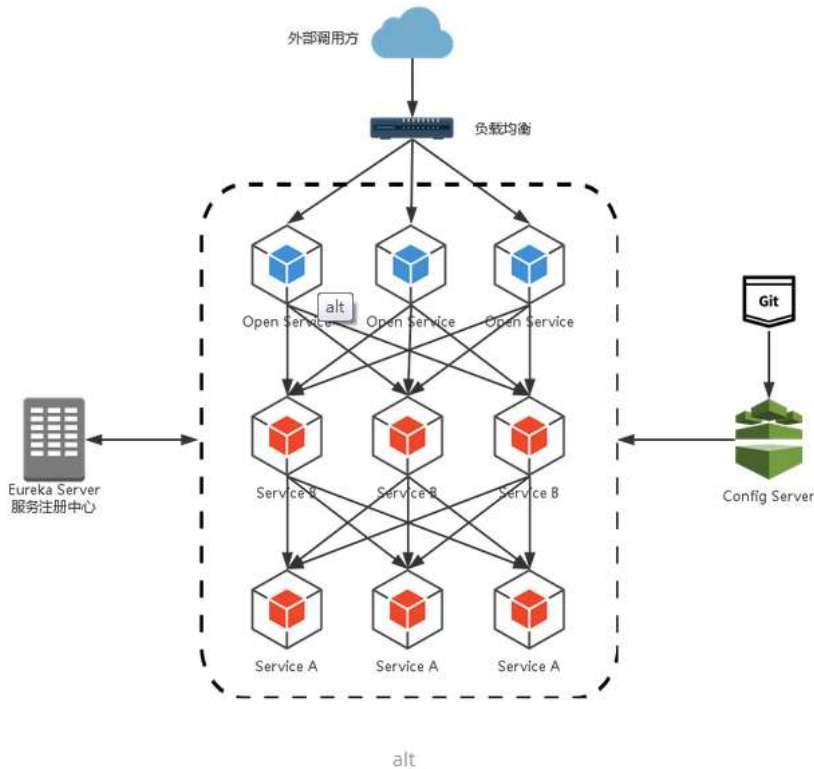
原 spring cloud ： 四、路由网关（zuul）

2018年03月08日 16:52:59

阅读数：177

前面几篇，简单总结了spring cloud应用中需要用到几个基础组件，包括服务的注册与发现（eureka），服务的消费（ribbon,feign），负载均衡，断路器（hystrix），这篇文章简单总结一下另一个比较重要的基础组件，路由网关（zuul），路由网关是什么？它能干什么？为什么要用它呢？

先上一张从网上找到的图，这张图的表示的是以微服务为基础的分布式系统的架构图，如果这幅图有点不明白，先往下看，稍后再回过头来看：



分布式架构的基础是微服务，所有的功能模块都会被抽成一个独立的微服务，所有的外部调用都要先经过一个总的入口，这个总的入口就相当于一道墙，把外部与内部的所有服务接口隔开，所有外部调用都要先经过这个总入口，由这个总的入口来决定，要调用哪个服务，这个总的入口就是路由网关，然后路由网关决定你需要调用哪些服务，内部的服务之间也是可以相互调用，如果有必要很多的服务会集群部署，包括总入口路由网关。有的人就会问了，为什么要在外部与内部所有的服务之间加上了这道墙呢？那是因如果让所有外部调用直接去调用自己所需要的服务接口，从安全性上来说，每个服务接口都要加上权限控制，这样一来权限控制的代码不可避免的会污染到服务接口的业务逻辑代码，其次每一个服务都加上权限控制，也造成了代码的重复，所以如果让外部调用直接调用内部服务接口会不可避免出现一系列的问题。

为了解决上面的这些问题，有人就提出一种解决问题的思路，就是在外部与内部的服务之间加上一个的总入口，把权限控制等这样重复性的功能抽离出来，放到这个总的入口出，这个的入口就是这篇文章要介绍的服务的路由网关。

因此对于分布式系统来说，服务网关是不可或缺的部分，它具备了服务路由、负载均衡、权限控制等功能，为内部的所有微服务提供了一个前门保护的作用，spring cloud netflix的zuul就担任了这样一个角色。

新建一个maven项目：serviceGateway

Pom.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.gaox.serviceGateway</groupId>
  <artifactId>serviceGateway</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>serviceGateway</name>
  <description>Demo project for Spring Boot</description>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>1.5.10.RELEASE</version>
    <relativePath/> <!-- lookup parent from repository -->
  </parent>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
    <java.version>1.8</java.version>
    <spring-cloud.version>Edgware.SR2</spring-cloud.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-starter-eureka</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-starter-zuul</artifactId>
    </dependency>

    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-test</artifactId>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>org.springframework.cloud</groupId>
        <artifactId>spring-cloud-dependencies</artifactId>
        <version>${spring-cloud.version}</version>
        <type>pom</type>
      </dependency>
    </dependencies>
  </dependencyManagement>
</project>
```



```

</dependencies>
</dependencyManagement>

<build>
<plugins>
<plugin>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>

</project>

```

在启动类ServiceGatewayApplication.java上加注解 @EnableZuulProxy，开启zuul功能

，这里加一个小技巧，@SpringCloudApplication相当于@SpringBootApplication、@EnableDiscoveryClient、@EnableCircuitBreaker三个功能，主要是简化配置。

```

package com.gaox.serviceGateway;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.netflix.eureka.EnableEurekaClient;
import org.springframework.cloud.netflix.zuul.EnableZuulProxy;

@SpringBootApplication
@EnableEurekaClient
@EnableZuulProxy
public class ServiceGatewayApplication {

    public static void main(String[] args) {
        SpringApplication.run(ServiceGatewayApplication.class, args);
    }
}

```

application.properties：

```

server.port=8766
spring.application.name=serviceGateWay
eureka.client.service-url.defaultZone=http://localhost:8761/eureka/
zuul.routes.zhangsanHello.path=/zhangsanHello/**
zuul.routes.zhangsanHello.serviceId=zhangsanService
zuul.routes.lisiHello.path=/lisiHello/**
zuul.routes.lisiHello.serviceId=lisiService

```

通过路由网关的功能，对外提供的服务只需要暴露zuul中配置的调用地址就可以让外部统一调用内部的服务，而不需要了解具体提供服务的主机信息了，zuul提供了两种的映射方式

第一种，也是比较推荐的一种，就是上面的配置，所有的内部服务都在eureka server服务注册中心注册，zuul的路由网关也注册到eureka server，所有服务之间可以彼此发现，这样以来zuul服务也就能在eureka server发现其他服务，我们就可以实现对serviceId的映射。

第二种就是如下配置

```

server.port=8766
spring.application.name=serviceGateWay
eureka.client.service-url.defaultZone=http://localhost:8761/eureka/
zuul.routes.zhangsanHello.path=/zhangsanHello/**
zuul.routes.zhangsanHello.serviceId=zhangsanService

```



```
zuul.routes.lisiHello.path=/lisiHello/**
zuul.routes.lisiHello.url=http://localhost:8765/
```

所有的符合/lisiHello/**的访问都会映射到<http://localhost:8765/>，也就是说当我们访问<http://localhost:8766/lisiHello/hello?name=lisi>的时候，zuul会将该请求路由到<http://localhost:8765/hello?name=lisi>上。

之所以不推荐这一种是因为如果内部的服务大多采用的是集群部署，我们需要知道所有的微服务的主要地址和端口，才能达到所有的映射配置，同时也失去了前端负载均衡的功能，不能根据实时情况合理分配请求到哪些服务。实际上，服务名与服务实例地址的关系在把服务在eureka server中注册后就已经存在了，我们只需要将zuul的服务也注册到eureka server中就可以发现其他服务，也就可以实现对serviceId的映射。

用第一种方法配置后，依次启动serviceCenter、helloService、zhangsanService、lisiService、serviceGateway，然后打开浏览器访问<http://localhost:8766/lisiHello/hello?name=lisi>

，第一次调用时可能会连接超时，后台报异常，不会没关系，多刷新两遍就会交替出现下面内容：

你好，lisi。我是helloService,端口是8763

你好，lisi。我是helloService,端口是8762

至于第一次为什么会报超时异常，以后再调用就存在异常的原因，一时还这不是很明白，这里推荐一篇博客，也许对我们有帮助：<https://juejin.im/entry/5a114ca6f265da4335625a36>

上面说了，对一些通用的控制可以抽离到路由网关里，这里用到的就是路由网关的过滤器，在完成了路由之后，通过过滤器来实现对服务的安全控制。在服务网关中定义过滤器只需要继承ZuulFilter抽象类并实现其定义四个抽象方法，以对请求进行拦截和过滤。

MyFilter.java：

```
package com.gaox.serviceGateway;

import com.netflix.zuul.ZuulFilter;
import com.netflix.zuul.context.RequestContext;
import org.apache.log4j.Logger;

import javax.servlet.http.HttpServletRequest;
import java.util.ArrayList;
import java.util.Enumuration;
import java.util.List;

/**
 * Created by Administrator on 2018/3/8.
 */
public class MyFilter extends ZuulFilter {
    private Logger logger=Logger.getLogger(MyFilter.class);

    @Override
    public String filterType() {
        return "pre";
    }

    @Override
    public int filterOrder() {
        return 0;
    }

    @Override
    public boolean shouldFilter() {
        return true;
    }

    @Override
    public Object run() {
        RequestContext rt= RequestContext.getCurrentContext();
        HttpServletRequest request=rt.getRequest();
        Enumuration<String> parameterName= request.getParameterNames();
        List<String> list=new ArrayList<>();
```

```
String name=parameterName.nextElement();
list.add(name);
System.out.println(name+":"+request.getParameter(name));
}
if(list.size()<=0){
    rt.setSendZuulResponse(false);
    rt.setResponseBody("At least one parameter");
}
return null;
}
}
```

在启动的时候在容器中实例化该过滤器，就可以起作用了：

```
@Bean
public MyFilter accessFilter(){
    return new MyFilter();
}
```

filterType：返回一个字符串代表过滤器的类型，在zuul中定义了四种不同生命周期的过滤器类型，具体：

pre：可以在请求被路由之前调用

routing：在路由请求时候被调用

post：在routing和error过滤器之后被调用

error：处理请求时发生错误时被调用

filterOrder：通过int值来定义过滤器的执行顺序

shouldFilter：返回一个boolean类型来判断该过滤器是否要执行，所以通过此函数可实现过滤器的开关。这里，我们直接返回true，所以该过滤器总是生效。

run：过滤器的具体逻辑。需要注意，这里我们通过rt.setSendZuulResponse(false)令zuul过滤该请求，不对其进行路由，然后通过rt.setResponseStatusCode(401)设置了其返回的错误码，当然我们也可以进一步优化我们的返回，比如，通过rt.setResponseBody(body)对返回body内容进行编辑等。，或者重新定向到一个错误页面

版权声明：本文为博主原创文章，未经博主允许不得转载。 <https://blog.csdn.net/fox9916/article/details/79486787>

个人分类：[spring cloud](#)

所属专栏：[spring cloud](#)

0

收藏

评论

微信

微博

QQ

[查看更多>>](#)


想对作者说点什么？ [我来说一句](#)

@ResponseBody 返回乱码 的完美解决办法 详解（值得收藏）

返回结果" ? ? " Java代码 @RequestMapping(value = "/getForm") @ResponseBody public String...

 jpr1990 2012-07-02 23:01:35 阅读量：19154

乱码的描述（废话）：中文都成了乱码。我在网上找了很多解决方案，但是基本上都没解决我的问题，直到翻了一下午的帖子找到了一个解决我项目乱码的一个解决方案。一般乱码问题基本上都是...

 zhanghaishan

2017-12-25 17:31:12

阅读数：1791

“性”生活太快？问题究竟出在哪里？

北京登蓝 · 顶新

Spring Cloud Config 快速入门

Spring Cloud Config分布式配置服务由服务器端和客户端共同组成，服务端提供配置信息的存储管理，对配置进行统一的管理配置文件，我们需要先把配置文件... 里，客户端读取配置信息 为了能统

 mn960mn

2016-06-29 10:57:45

阅读数：6138

SpringCloud上传文件，经过Zuul，中文文件名乱码解决办法

最近在使用SpringCloud搭建微服务的过程中，发现上传文件经过Zuul网关转发的时候，回应为上传文件名中文乱码导致文件的写操作失败，从而导致文件上传失败，但是不经过Zuul转发的时候，文件上传正...

 zj20142213

2018-03-17 09:38:27

阅读数：517

spring cloud中返回数据编码问题

spring cloud集成了spring mvc，默认返回的格式都是json，utf-8。如果在项目中发现返回的json失败，或者编码有问题，需要检查一下是否做了以下配置：
@Conf...

 lovebird321

2017-12-19 11:24:15

阅读数：612

Spring boot 入门之Zuul

1.创建spring boot基础工程，

 qinqigang

2017-09-19 17:03:13

阅读数：394

痔疮这么简单就能好？早知道就不割痔疮了！

广医德 · 顶新

史上最简单的SpringCloud教程 | 第五篇: 路由网关(zuul)

转载请标明出处： <http://blog.csdn.net/forezpf/article/details/69939114> 本文出自方志朋的博客 在微服务架构中，需要几个基础的服务治...

 juan0728juan

2017-10-12 15:34:20

阅读数：184

zuul上传中文文件乱码解决.

项目中用到zuul，不起网关上传文件不会出现中文乱码，经过网关处理后，中文都变成??.查了好久的解决方案都是在url加前缀/zuul来访问。这样确实可以解决问题。但是项目很多定义给客户端访问的api都...

 wsmlhxt777

2018-01-29 10:16:34

阅读数：359

Spring乱码问题解决方案

请求乱码 GET请求乱码： 原因：请求参数带在url地址上、url地址什么时候解析？ tomcat收到请求对url进行编解码（ISO8859-1） 解决方案：在tomcat的8080端口配置出...

 ggabcda

2017-08-16 21:33:28

阅读数：81

关于spring boot属性文件中文参数文输出到页面乱码

在eclipse中，以.properties后缀的属性文件默认是ISO-8859-1,即字节编码,这种编码方式向下兼容ASCII，而Unicode编码与ISO-8859-1标准兼容，UTF-8则是对U...

 fox9916

2017-10-24 17:17:39

阅读数：578

spring服务器乱码问题

ajax请求： 1)ajax请求 ajax请求不会使用 meta http-equiv="Content-Type" content="text/html; charset=utf-8" />的编码

0

收藏

评论

微信

微博

QQ

SpringCloud教程 | 六.路由网关zuul

简介 在微服务架构中，需要几个基础的服务治理组件，包括服务注册与发现、服务消费、负载均衡、断路器、智能路由、配置管理等，由这几个基础组件相互协作，共同组建了一个简单的微服务系统。在Spring...

 gwd1154978352 2018-01-26 14:08:38 阅读数：128

spring cloud之简单介绍

以下是来自官方的一篇简单介绍：Spring Cloud provides tools for developers to quickly build some of the common pat...


 minicto 2016-11-28 14:15:25 阅读数：65029

程序猿速成英语秘籍→

老司机教你一个数学公式秒懂天下英语

史上最简单的SpringCloud教程 | 第五篇: 路由网关(zuul)

在微服务架构中，需要几个关键的组件，服务注册与发现、服务消费、负载均衡、断路器、智能路由、配置管理等，由这几个组件可以组建一个简单的微服务架构。客户端的请求首先经过负载均衡（zuul、Ngnix），再...

 forezp 2017-04-09 23:25:01 阅读数：261756

Spring Cloud微服务断路器Hystrix与服务网关学习笔记

上一篇文章介绍了如何使用Spring Cloud与Spring Boot以及Docker容器搭建一个微服务模式的示例(<https://github.com/foxcpp/jianGo/SpringCloudL...>)

 canot 2017-03-20 23:51:18 阅读数：5859

Springcloud学习——Zuul服务网关及路由权限控制

Springcloud使用zuul服务网关

 qq_19301269 2017-12-28 12:39:16 阅读数：2170

一起来学SpringCloud之 - 路由网关（Zuul）

上一篇已经讲了微服务组件中的 分布式配置中心，本章讲述 由JAVA编写的服务路由网关Zuul.....

 memmsc 2017-09-28 14:01:08 阅读数：2429

Spring Cloud Zuul实现动态路由（zuul设计）

转载来源： <https://www.wxzhi.com/archives/813/tn2h50sbyqjrp1e3/> 前言 Zuul 是Netflix 提供的一个开源组件,致力于在云平台上...

 jackphang 2017-11-04 09:50:12 阅读数：1973

免费云主机试用一年

云主机免费推荐吗

百度广告



spring cloud之zuul路由网关（六）

转自： <http://blog.csdn.net/forezp/article/details/69939114>在微服务架构中，需要几个基础的服务治理组件，包括服务注册与发现、服务消费、负载均衡、断...

 u014401141 2017-11-28 17:34:08 阅读数：371

Spring Cloud Zuul--服务网关

有了Eureka服务注册发现、Hystrix断路器、Ribbon服务调用负载均衡，以及spring cloud config 集群配置中心，似乎一个微服务框架已五脏俱全，last but not l...

 Daybreak1209 2016-12-19 10:41:56 阅读数：32024

Spring Cloud（六）服务网关 zuul 快速入门

服务网关是微服务架构中一个不可或缺的部分。通过服务网关统一向外系统提供REST AP的过程中，除了具备服务路由、均衡负载均衡功能之外，它还具备了权限...

 0

 收藏

 评论

 微信

 微博

 QQ



ianGo/SpringCloudL...

SpringCloud系列：利用Zuul实现统一服务网关服务，简单实现IP白名单功能

本文基于SpringCloud Zuul实现统一网关服务、请求校验等功能，最后通过zuul过滤器实现IP白名单功能...

tjeagle 2017-12-13 17:25:41 阅读数：2984

Spring Cloud_24_网关Zuul/路由配置

网关Zuul/路由配置 Zuul自带负载均衡功能，何时使用负载均衡，何时使用普通转发？ 1、简单路由 SimpleHostRoutingFilter 配置连接池 zuul.host.ma...

zhaozao5757 2018-03-21 09:32:21 阅读数：10237

高效、准确、低成本的文字识别api

证件、文件一键扫描，准确率高达99%。经过海量用户和多种复杂场景考验



springcloud zuul gateway 服务网关

2017年02月21日 19KB 下载

第二十三章 使用Spring cloud+zuul使用API网关进行上传文件

使用Spring cloud+zuul使用API网关进行上传文件 项目结构如下： ZuulUploadController类 package com.example.de tr...

qq_24504315 2018-01-22 20:48:43 阅读数：622

Spring Cloud Zuul网关 Filter、熔断、重试、高可用的使用方式。

时间过的很快，写springcloud(十)：服务网关zuul初级篇还在半年前，现在已经是2018年了，我们继续探讨Zuul更高级的使用方式。 上篇文章主要介绍了Zuul网关使用模式，以及自动转发机...

ityouknow 2018-01-31 12:25:21 阅读数：1953

Spring Cloud构建微服务架构（五）服务网关

引入依赖spring-cloud-starter-zuul、spring-cloud-starter-eureka，如果不是通过指定serviceld的方式，eureka依赖不需要，但是为了对服务集群...

qq_25353539 2016-08-03 11:46:40 阅读数：5214

spring cloud zuul网关服务重试请求配置和源码分析

我们一般部署服务的时候，都会部署一个网关服务，内部所有的其他微服务的调用，都将通过网关路由过去，不对外直接暴露，对外只暴露网关服务。而且一般内部服务会部署多个实例，zuul集成了ribbon，会自动负...

xiao_jun_0820 2018-02-13 12:06:02 阅读数：474

免费云主机试用一年

免费云主机试用一年厂家批发

百度广告



springcloud入门之路由网关zuul

在前几篇springcloud的学习中，我们分别尝试搭建了一个分布式项目架构的基础模块，使用eureka实现服务的注册与发现的服务注册中心eureka-sercer和服务提供者eureka-clien...

chenhaotao 2017-12-05 17:29:39 阅读数：120

spring cloud网关zuul源码之Filter管理

接上一篇request管理。 原文地址：http://blog.csdn.net/haha7289/article/details/54344150 zuul支持动加载Filter类文件。...

tianyaleixiaowu 2017-04-11 11:12:12 阅读数：910

SpringCloudAPI网关服务Zuul

SpringCloud微服务就是把一个大的项目拆分成多个小的模块，然后模块之间通过远程调用、服务治理的技术互相配合工作，随着业务的增加，项目也将会越来越庞大，接口数量也随之增加，对外提供服务的接口也会...



【Spring Cloud】Zuul 路由访问

本节大纲： - 简介 - 架构图 - 创建项目（ 接着上一章节 ） - 在Fcat实战项目中的应用 简介zuul是Netflix出品的一个基于JVM路由和服务端的负载均衡器。类...

xf1195718067 2017-11-26 16:09:26 阅读数：187

SpringCloud教程十：Zuul+Mysql实现动态路由

实际开发中随着服务的增多，通过配置来设置zuul的路由就会变的很麻烦，给运维带来很大的不变，这里我们通过数据库获取的方式来实现动态路由的配置：首先sql:CREATE TABLE `gateway_a...

wang_shuyu 2018-02-10 13:41:51 阅读数：620

高效、准确、低成本的文字识别api

证件、文件一键扫描，准确率高达99%。经过海量用户和多种复杂场景考验



SpringCloud学习笔记-zuul网关

SpringCloud zuul网关和网络拓补图结构，讲解路由和拦截功能

brycegao321 2017-11-28 13:10:56 阅读数：618

Spring基础：快速入门spring cloud（4）：API网关之Zuul

Spring Cloud是Spring总多的Project中的一个,它提供了一整套的工具帮助系统架构师们在进行分布式设计的时候可以...用,在创建和发布微服务时极为便捷和有效。本系列文章将会使用最...

liumiaocn 2016-12-30 12:25:39 阅读数：21645

spring cloud 入门系列六：使用Zuul 实现API网关服务

通过前面几次的分享，我们了解了微服务架构的几个核心设施，通过这些组件我们可以搭建简单的微服务架构系统。比如通过Spring Cloud Eureka搭建高可用的服务注册中心并实现服务的注册和发现；通过...

shunfa888 2018-05-09 00:19:46 阅读数：16

Spring Cloud微服务（3）之网关Zuul

1.介绍 古语有云：一夫当关，万夫莫开。网关Zuul就在系统中起到同样的作用，它是系统的门户、城市的关隘、公园的检票口。服务网关API Gateway可以有...很多实现方法，如Nginx、Zuul，甚...

u013084910 2017-07-30 14:58:46 阅读数：1732

Spring Cloud Zuul微服务网关的API限流

API限流微服务开发中有时需要对API做限流保护，防止网络攻击，比如做一个短信验证码API，限制客户端的请求速率能在一定程度上抵御短信轰炸攻击，降低损失。微服务网关是每个请求的必经入口，非常适合做一些...

ta_ab 2017-09-14 19:32:40 阅读数：3771

免费云主机试用一年

可以免费试用30天的移动云vps

百度广告



springcloud---Zuul动态路由

前言Zuul 是Netflix 提供的一个开源组件,致力于在云平台上提供动态路由，监控，弹性，安全等边缘服务的框架。也有很多公司使用它来作为网关的重要组成部分，碰巧今年公司的架构组决定自研一个网关产品...


u013815546 2017-04-01 14:11:25 阅读数：26856

Spring Cloud Zuul（服务网关）

1.zuul的作用 1.1服务的统一入口 Zuul的主要功能是路由和过滤器。是各种服务的统一入口，同时还会用来提供监控、授权、安全、调度等等。如上图所示，在无网关的情况下，...

jack281706 2017-06-26 18:01:36 阅读数：2088

个人资料



冰与火IAF

关注

原创21

粉丝9

喜欢7



评论9


等级：博客已

访问：2万+







积分：414

排名：12万+

勋章： 



墙体彩绘机



最新文章

关于springboot+shiro+thymeleaf页面级元素的权限控制的问题


Spring +mybatisplus+shiro权限管理集成整合

spring cloud：五、分布式配置中心（spring cloud config）

spring cloud：三、断路器（hystrix）

spring cloud：二、服务的消费（ribbon和feign两种方式）

博主专栏



spring cloud

阅读量：124365篇

个人分类

spring3篇

Java基础5篇

database1篇

git2篇

springboot4篇

展开

归档

2018年5月2篇

2018年3月6篇

2018年2月2篇

2017年10月5篇

2017年7月1篇



0



收藏



评论



微信



微博



QQ

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！

登录注册

×

https://blog.csdn.net/fox9916/article/details/7948678710/11

热门文章

- [spring cloud：五、分布式配置中心（spring cloud config）](#)
阅读量：11397
- [git本地项目代码上传至码云远程仓库总结](#)
阅读量：3891
- [spring webflow 学习小结](#)
阅读量：3630
- [关于spring boot属性文件中中文参数文输出到页面乱码](#)
阅读量：573
- [spring cloud：一、服务的注册与发现](#)
阅读量：268

最新评论

- [spring cloud：五、分...](#)
u010897136：....
- [spring webflow 学习...](#)
fox9916：[reply]qq_24646129[reply]能跑通的
- [spring cloud：五、分...](#)
fox9916：[reply]qq_31559209[reply]git的用户名密码是需要配置的
- [spring cloud：五、分...](#)
qq_31559209：问下我在将分布式配置中心加入到eurake中后在客户端就读取不到git上的配置属性了是怎么回事。
- [spring webflow 学习...](#)
qq_24646129：感觉zhengzhou.xml，web-application-config.xml这些文件都不全啊...



联系我们

请扫描二维码联系客服

✉ webmaster@csdn.net

☎ 400-660-0108

👤 QQ客服 💬 客服论坛

[关于](#) [招聘](#) [广告服务](#) [百度](#)

©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

- 0
- 收藏
- 评论
- 微信
- 微博
- QQ