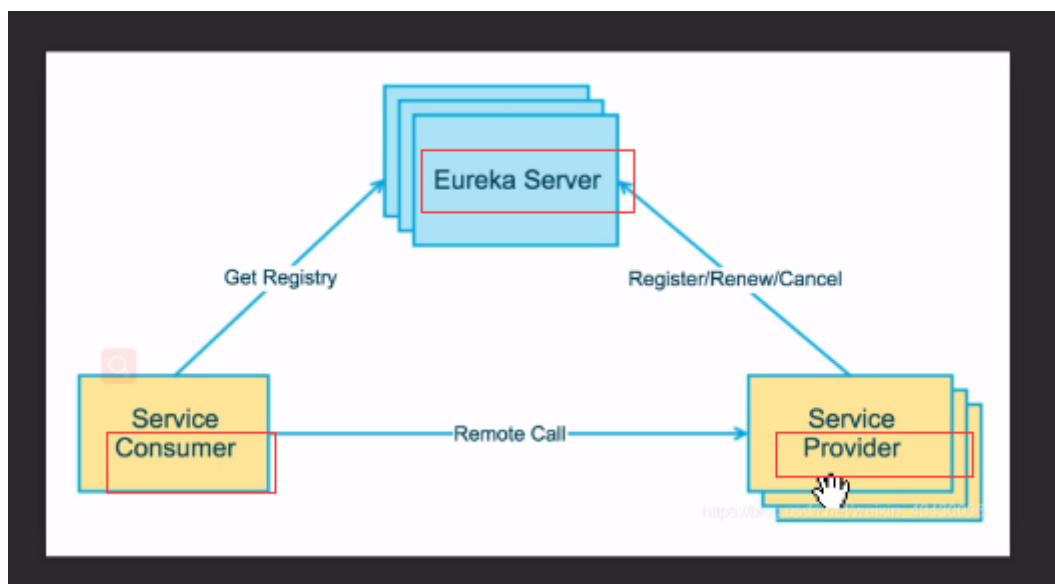


一.springcloud学习

Netflix公司的

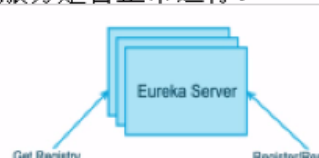


解决的问题:服务比较多时,我们可以在注册中心通过服务名调用

```
portal >>>rest 服务
1: 直接url 调用
存在问题: rest服务的服务器ip发生改变, 或者端口发生改变
假如说有五百个服务? 都发生IP改变
以上这种方式: 硬编码的调用

springcloud eureka

把消费者和生产者这些服务, 都注册到eureka注册中心里面
, 可以通过服务名调用
服务名在这里面代表http://ip+端口
```

boot springcloud 父工程的搭建		1 eureka 服务中心的创建	2 sso-rest的eureka客户端的创建
D	E		
第1天	springboot springcloud 父工程的搭建 1 eureka 服务中心的创建 2 sso-rest的eureka客户端的创建		引入的是spring-cloud-starter-net @EnableEurekaServer 引入的是spring-cloud-starter-net @EnableEurekaClient
第2天	Eureka Server 作为服务注册功能的服务器，它是服务注册中心。而系统中的其他微服务，使用 Eureka 的客户端连接到 Eureka Server，并维持心跳连接。这样系统的维护人员就可以通过 Eureka Server 来监控系统中各个微服务是否正常运行。		eureka由两个组件组成：Eureka Server用作服务注册服务器。Eureka 化与服务器的交互、作为轮询负载均衡 Netflix在其生产环境中使用的是另外 利用率以及出错状态的加权负载均衡
			Eureka Server(注册中心):提供服务 Service Provider(服务提供方):将自

二.项目的使用:

1.创建maven项目,父工程(springcloud-parent),pom文件,只要架包

```

1 <modelVersion>4.0.0</modelVersion>
2 <groupId>com.lanou</groupId>
3 <artifactId>springcloud-parent</artifactId>
4 <version>0.0.1-SNAPSHOT</version>
5 <packaging>pom</packaging>
6
7 <properties>
8 <springboot.version>2.1.3.RELEASE</springboot.version>
9 <springcloud.verison>Greenwich.RELEASE</springcloud.verison>
10 </properties>
11
12 <dependencyManagement>
13 <!-- springboot -->
14 <dependencies>
15 <dependency>
16 <groupId>org.springframework.boot</groupId>
17 <artifactId>spring-boot-dependencies</artifactId>
18 <version>${springboot.version}</version>
19 <type>pom</type>
20 <scope>import</scope>
21 </dependency>
22 <dependency>

```

```
23 <groupId>org.springframework.cloud</groupId>
24 <artifactId>spring-cloud-dependencies</artifactId>
25 <version>${springcloud.verison}</version>
26 <type>pom</type>
27 <scope>import</scope>
28 </dependency>
29 </dependencies>
30 <!-- springcloud -->
31 </dependencyManagement>
32 </project>
```

注意:springcloud和springboot保持一致版本号.

The screenshot shows the Spring Cloud website interface. The top navigation bar includes the Spring logo, 'PROJECTS', 'GUIDES', 'BLOG', and 'TRAINING & CERTIFICATION'. The left sidebar lists various projects, with 'Spring Cloud' highlighted and expanded. The main content area is titled 'Spring Cloud' with a 'Greenwich' version indicator. Below this, a sub-navigation bar has 'Overview', 'Learn', and 'Samples' (which is selected and highlighted with a red box). The 'Samples' section lists several examples to try out, including 'Config Server', 'Service Registry', 'Circuit Breaker Dashboard', 'Business Application (Customers and Stores)', 'OAuth2 Authorization Server', 'OAuth2 SSO Client', 'Integration Test Samples', and 'Spring Cloud Contract Samples'.

Boot compatibility	
Release Train	Boot Version
Greenwich	2.1.x
Finchley	2.0.x
Edgware	1.5.x
Dalston	1.5.x

Table 2. Release train contents

Component	Edgware.SR5	Finchley.SR2	Finchley.BUILD-SNAPSHOT
spring-cloud-aws	1.2.3.RELEASE	2.0.1.RELEASE	2.0.1.BUILD-SNAPSHOT
spring-cloud-bus	1.3.3.RELEASE	2.0.0.RELEASE	2.0.1.BUILD-SNAPSHOT
spring-cloud-cli	1.4.1.RELEASE	2.0.0.RELEASE	2.0.1.BUILD-SNAPSHOT
spring-cloud-commons	1.3.5.RELEASE	2.0.2.RELEASE	2.0.2.BUILD-SNAPSHOT
spring-cloud-contract	1.2.6.RELEASE	2.0.2.RELEASE	2.0.2.BUILD-SNAPSHOT
spring-cloud-config	1.4.5.RELEASE	2.0.2.RELEASE	2.0.2.BUILD-SNAPSHOT
spring-cloud-netflix	1.4.6.RELEASE	2.0.2.RELEASE	2.0.2.BUILD-SNAPSHOT
spring-cloud-security	1.2.3.RELEASE	2.0.1.RELEASE	2.0.1.BUILD-SNAPSHOT
spring-cloud-cloudfoundry	1.1.2.RELEASE	2.0.1.RELEASE	2.0.1.BUILD-SNAPSHOT
spring-cloud-consul	1.3.5.RELEASE	2.0.1.RELEASE	2.0.2.BUILD-SNAPSHOT

2.新建maven项目,jar形式,需要手动导入springboot,继承于父工程

```

1 <modelVersion>4.0.0</modelVersion>
2 <!-- 继承于父工程 -->
3 <parent>
4 <groupId>com.lanou</groupId>
5 <artifactId>springcloud-parent</artifactId>
6 <version>0.0.1-SNAPSHOT</version>
7 </parent>
8 <groupId>com.lanou</groupId>
9 <artifactId>eureka-server</artifactId>
10 <version>0.0.1-SNAPSHOT</version>
11
12 <dependencies>
13 <!-- sprig-boot -->
14 <dependency>

```

```
15 <groupId>org.springframework.boot</groupId>
16 <artifactId>spring-boot-starter-web</artifactId>
17 </dependency>
18 <dependency>
19 <groupId>org.springframework.boot</groupId>
20 <artifactId>spring-boot-devtools</artifactId>
21 </dependency>
22 <dependency>
23 <groupId>org.springframework.boot</groupId>
24 <artifactId>spring-boot-starter-test</artifactId>
25 <scope>test</scope>
26 </dependency>
27
28 <!-- springcloud -->
29 <!-- 只要注册中心才用到该依赖 -->
30 <dependency>
31 <groupId>org.springframework.cloud</groupId>
32 <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>
33 </dependency>
34 <!-- 为eureka添加安全机制 -->
35 <dependency>
36 <groupId>org.springframework.boot</groupId>
37 <artifactId>spring-boot-starter-security</artifactId>
38 </dependency>
39 </dependencies>
40
41 <build>
42 <plugins>
43 <plugin>
44 <groupId>org.springframework.boot</groupId>
45 <artifactId>spring-boot-maven-plugin</artifactId>
46 <configuration>
47 <fork>true</fork>
48 </configuration>
49 </plugin>
50 </plugins>
51
52 </build>
```

```
53 </project>
```

3.需要写启动类,和yml语言,yml如下:

```
1 server:
2   port: 12000
3
4 spring:
5   security:
6     user:
7       name: admin
8       password: admin
9
10
11
12 eureka:
13   instance:
14     hostname: localhost
15   client:
16     register-with-eureka: false
17     fetch-registry: false
18     service-url:
19       defalutZone: http://admin:admin@${eureka.instance.hostname}:${server.port}/eureka/
```

4.安全问题:可设置springcloud账号,密码.

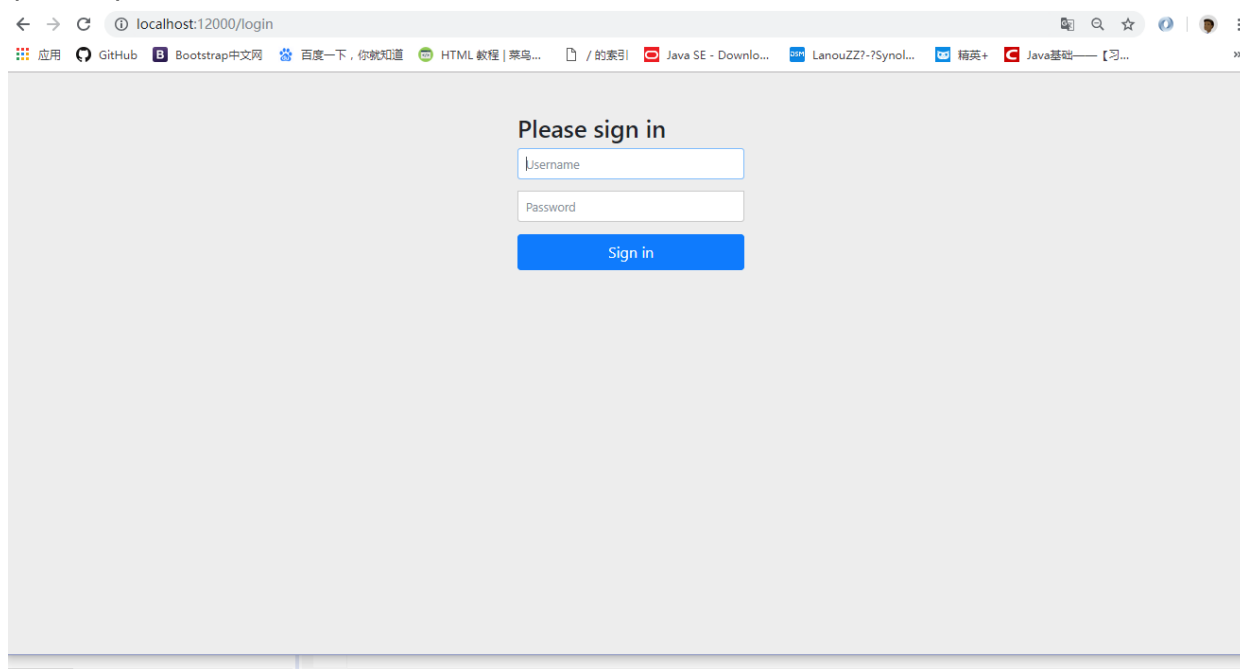
```
1 <!-- 为eureka添加安全机制 -->
2 <dependency>
3   <groupId>org.springframework.boot</groupId>
4   <artifactId>spring-boot-starter-security</artifactId>
5 </dependency>
```

同样在yml语言中,改成以上的形式.注:yml语言中的引用可以用
@\${eureka.instance.hostname}引用.


```
eureka-server/pom.xml  EurekaServerApp.java  application.yml
1 server:
2   port: 12000
3
4 eureka:
5   instance:
6     hostname: localhost
7   client:
8     register-with-eureka: false
9     fetch-registry: false
10    service-url: http://localhost:12000/eureka/
```

5.页面如下:

(安全性)



(进入页面)

 **spring Eureka**

HOME LAST 1000 SINCE STARTUP

System Status

Environment	test	Current time	2019-04-15T11:49:38 +0800
Data center	default	Uptime	00:01
		Lease expiration enabled	false
		Renews threshold	1
		Renews (last min)	0

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
No instances available			

General Info

Name	Value
total-avail-memory	418mb
environment	test
num-of-cpus	4
current-memory-usage	74mb (17%)

管理的端口号.

