好的，非常感谢您提供详细的配置文件！这是解决问题的关键。

根据您给出的 zuul-service 配置，我们之前的验证URL确实会产生404错误。问题主要出在 zuul.prefix 和自定义的 path 上。您的配置文件写得非常规范，现在我们来精确地解读它，并生成一份**绝对正确**的验证手册。

### **核心配置解读**

在开始验证前，我们先快速看一下 zuul-service 的 application.yml 中的三个关键配置，这能帮助我们理解URL是如何构成的：

1. **zuul.prefix: /api**
   * 这是一个**全局前缀**。所有通过Zuul网关的请求，都必须以 /api 开头。这是导致之前404错误的**最主要原因**。
2. **hello-server 的路由规则**  
   YAML  
   hello-server:  
    path: /server/\*\*  
    url: http://localhost:8071  
   * 这是一个**静态URL路由**。它告诉 Zuul：
     + 当一个请求路径匹配 /server/\*\* 时，应用此规则。
     + 将请求直接转发到固定的 http://localhost:8071 地址。
     + 这个过程**不依赖 Eureka**。
3. **hello-client 的路由规则**  
   YAML  
   hello-client:  
    path: /client/\*\*  
    serviceId: HELLO-CLIENT  
   * 这是一个**服务发现路由**。它告诉 Zuul：
     + 当一个请求路径匹配 /client/\*\* 时，应用此规则。
     + 去 Eureka 注册中心查找名为 HELLO-CLIENT 的服务。
     + 将请求动态转发到该服务的一个实例上（即 http://localhost:8072）。

### **修正后的精确验证手册**

请确保您的 EurekaServiceApplication(8070), HelloServerApplication(8071), HelloClientApplication(8072), 和 ZuulServiceApplication(8079) 都已启动。

#### **第一步：验证 Eureka 服务注册中心**

这一步保持不变，用于确认后台服务已准备就绪。

1. 打开浏览器，访问 Eureka 管理界面：  
   http://localhost:8070
2. 预期结果：  
   在 "Instances currently registered" 部分，您应该能看到 HELLO-SERVER 和 HELLO-CLIENT 这两个服务。

#### **第二步：验证 Zuul 网关路由（正确的方式）**

现在，我们将使用结合了**全局前缀**和**自定义路径**的正确URL来访问服务。

**A) 验证 hello-server 路由 (静态URL方式)**

1. 打开浏览器，访问以下地址：

例子：  
<http://localhost:8079/api/server/hello>

实际：  
[http://localhost:8079/api/server/rest/hello/server](http://localhost:8079/api/server/hello)

1. **URL 构成解析**:
   * http://localhost:8079：Zuul 网关的地址。
   * /api：来自 zuul.prefix 的**全局前缀**。
   * /server/：匹配 hello-server 规则中的 path: /server/\*\*。
   * hello：这是 hello-server 微服务自身暴露的端点路径（假设为 /hello）。

**B) 验证 hello-client 路由 (服务发现方式)**

1. 打开浏览器，访问以下地址：

例子：  
http://localhost:8079/api/client/hello

实际：

http://localhost:8079/api/client/rest/hello/client

1. **URL 构成解析**:
   * http://localhost:8079：Zuul 网关的地址。
   * /api：来自 zuul.prefix 的**全局前缀**。
   * /client/：匹配 hello-client 规则中的 path: /client/\*\*。
   * hello：这是 hello-client 微服务自身暴露的端点路径（假设为 /hello）。

预期结果：

这次访问上述两个 URL，您应该都能看到来自对应服务的成功响应，不再是 404 错误。

#### **第三步：检查 Zuul 的路由表 (高级验证)**

这个步骤依然有效，可以帮助我们确认 Zuul 内部的路由规则是否正确加载。

1. 打开浏览器，访问：  
   http://localhost:8079/manage/routes
2. 预期结果：  
   您会看到一个 JSON 输出，其中包含了 Zuul 解析后的路由信息。您应该能清晰地看到类似下面的条目，确认您的自定义 path 已经生效：
   * /api/server/\*\*: hello-server
   * /api/client/\*\*: hello-client

{"/api/server/\*\*":"http://localhost:8071",

"/api/client/\*\*":"HELLO-CLIENT",

"/api/hello-client/\*\*":"hello-client",

"/api/hello-server/\*\*":"hello-server"}

### **为什么之前会404？**

根据您的配置文件，导致404错误的原因有两个：

1. **缺少全局前缀**：所有请求都必须加上 /api 前缀，我们之前的URL (/hello-server/hello) 缺少了这个部分。
2. **路径被自定义**: 您通过 path 属性为每个服务指定了**别名** (/server/ 和 /client/)。因此，不能再使用默认的服务名 (/hello-server/) 作为路径了。

请按照上面 **“修正后的精确验证手册”** 中的URL进行尝试，这次一定可以成功访问了！