**Spring Cloud Zuul 组件\_1.4.7版本\_部署手册（V1.0）**

**基础技术中心**

**2025年8月**

# 修订记录

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **编号** | **日期** | **描述** | **版本** | **作者** | **审核** | **发布日期** |
| 1 | 2025-08-13 | 建立文档 | 1.0 |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |

如无中国建设银行的书面许可，任何人都无权复制或利用。

**目 录**

[1. 修订记录 2](#_Toc5476)

[2. 适用范围 4](#_Toc20025)

[3. 安装包准备及下载链接 4](#_Toc19755)

[4. 部署要求 4](#_Toc11828)

[4.1. 部署原则 4](#_Toc8942)

[4.2. 资源需求 4](#_Toc10475)

[4.3. 部署规模 5](#_Toc10084)

[4.4. 操作系统要求 5](#_Toc985)

[5. 安装步骤 6](#_Toc2894)

[6. 组件参数配置 7](#_Toc28961)

[7. 启动与验证 8](#_Toc4867)

[8. 常见部署问题及解决 8](#_Toc7034)

[9. 附录 8](#_Toc16178)

# 适用范围

本手册旨在指导运维和开发人员在生产环境Linux操作系统上部署、配置和验证Spring Cloud Zuul 1.4.7网关应用。部署目标是完成高可用集群的搭建与基础配置，以确保网关服务的稳定、可靠运行。

# 安装包准备及下载链接

Spring Cloud Zuul作为微服务网关组件，通常以Spring Boot应用的形式进行打包。

* **安装包类型：** 可执行JAR包 (.jar) 或 WAR包 (.war)
* **内部仓库地址：** [内部Maven/Nexus/Artifactory仓库地址，例如：[http://nexus.yourcompany.com/repository/releases/](https://www.google.com/search?q=http://nexus.yourcompany.com/repository/releases/)...]
* **获取方式：**
  + 从持续集成（CI）流水线中获取最终构建产物。
  + 从内部Maven仓库下载指定版本的JAR包。

# 部署要求

该章节在部署前必须满足的条件，比如操作系统要求、磁盘大小、内存、网络连通性、其他软件依赖（如Java版本要求）。

## **部署原则**

* **高可用性：** 生产环境必须采用集群部署方式，并遵循反亲和性原则，将网关实例部署在不同的物理机、虚拟机或可用区（AZ）中，避免单点故障。
* **自动化：** 推荐使用自动化部署工具（如Ansible, Jenkins等）进行批量部署和配置管理。
* **资源隔离：** 网关应用应部署在独立的服务器或容器中，避免与其他关键应用共用资源。

## **资源需求**

根据预期流量和并发量进行调整，以下为建议的初始配置：

* **操作系统：** CentOS 7.x / Red Hat Enterprise Linux 7.x 或更高版本。
* **CPU：** 4核+
* **内存：** 8GB+
* **磁盘空间：** 20GB+ (用于存储日志、应用包等)
* **端口要求：** 8080 (应用端口)，其他端口根据具体配置。
* **必备软件：**
  + JDK 1.8 或更高版本（推荐使用Open JDK 8）。

## **部署规模**

为了确保服务的稳定与可靠性，建议至少部署3个以上的网关实例组成集群，并由前端负载均衡器（如Nginx, F5）进行流量分发。

# 安装步骤

1. **创建应用用户和目录**
   * 以root用户登录服务器。
   * 创建专用用户和用户组：useradd -m zuul
   * 创建应用部署目录：mkdir -p /app/zuul/
   * 将目录权限赋给zuul用户：chown -R zuul:zuul /app/zuul/
2. **上传安装包**
   * 将预先准备好的Zuul网关JAR包上传至 /app/zuul/ 目录。
3. **创建启动脚本**
   * 在 /app/zuul/ 目录下创建 start.sh 脚本，内容示例见附录A。
4. **配置系统服务（可选，推荐）**
   * 在 /etc/systemd/system/ 目录下创建 zuul-gateway.service 文件，内容示例见附录B。
   * 重新加载systemd配置：systemctl daemon-reload
   * 设置开机自启：systemctl enable zuul-gateway.service

# 组件参数配置

在 application.yml 或 application.properties 中进行核心配置，示例见附录C。

# 启动与验证

1. **启动服务**
   * 如果配置了systemd服务，执行：systemctl start zuul-gateway.service
   * 否则，手动执行启动脚本：sh /app/zuul/start.sh
2. **验证步骤**
   * **检查进程：** ps -ef | grep zuul-gateway，确认应用已启动。
   * **检查日志：** tail -f /app/zuul/logs/zuul.log，确认无报错信息。
   * **健康检查：** 访问 http://[网关IP]:[网关端口]/actuator/health，确认返回 "status": "UP"。
   * **路由验证：** 访问一个配置好的路由接口，例如 http://[网关IP]:[网关端口]/api/service-a/your-endpoint，确认请求能够正确转发到后端服务。

# 常见部署问题及解决

| **序号** | **问题描述** | **潜在原因** | **解决方法** |
| --- | --- | --- | --- |
| 1 | 启动失败，端口被占用 | server.port 指定的端口被其他进程占用。 | 使用 netstat -tulnp | grep [端口号] 查找占用进程，并修改端口或结束占用进程。 |
| 2 | 应用启动后无法被Eureka发现 | spring.application.name 与Eureka配置不一致，或网络不通。 | 检查application.yml配置；使用 ping 或 telnet 验证与Eureka服务器的网络连通性。 |
| 3 | 路由不生效，访问404 | 路由配置错误，或下游服务未在Eureka注册。 | 检查 zuul.routes 配置是否正确，确保服务ID或URL准确无误；检查下游服务在Eureka的状态。 |
| 4 | CPU或内存使用率过高 | 并发量过高，或JVM参数设置不当。 | 调整 -Xms 和 -Xmx 参数；使用Jvisualvm等工具分析内存/线程状况，必要时进行扩容。 |
| 5 | 请求超时，返回504/500 | Ribbon或Hystrix超时时间设置过短，或下游服务响应过慢。 | 适当增加 ribbon.ReadTimeout 和 hystrix.command.default.execution.isolation.thread.timeoutInMilliseconds 的值；排查下游服务性能问题。 |

# 附录

## 附录A：Zuul网关启动脚本示例

Bash  
#!/bin/bash  
JAVA\_HOME="/usr/local/java/jdk1.8.0\_201" # 替换为实际的JDK路径  
APP\_NAME="zuul-gateway-1.4.7.jar" # 替换为实际的JAR包名称  
LOG\_DIR="/app/zuul/logs"  
  
# 创建日志目录  
if [ ! -d $LOG\_DIR ]; then  
 mkdir -p $LOG\_DIR  
fi  
  
nohup $JAVA\_HOME/bin/java -jar -Xms4g -Xmx6g $APP\_NAME > $LOG\_DIR/zuul.log 2>&1 &  
echo "Zuul Gateway is starting..."

## 附录B：zuul-gateway.service 文件内容示例示例

Ini, TOML  
[Unit]  
Description=Zuul Gateway Service  
After=network.target  
  
[Service]  
User=zuul  
Group=zuul  
WorkingDirectory=/app/zuul  
ExecStart=/bin/sh /app/zuul/start.sh  
ExecStop=/bin/kill -9 $MAINPID  
Restart=always  
  
[Install]  
WantedBy=multi-user.target

## 附录C：application.yml 核心配置示例

# 应用基本信息  
spring:  
 application:  
 name: zuul-gateway # 服务注册到Eureka的名称  
   
# Zuul路由配置  
zuul:  
 prefix: /api # 全局统一前缀  
 routes:  
 service-a:  
 path: /service-a/\*\*  
 serviceId: service-a-app # 路由到服务注册中心的服务ID  
 service-b:  
 path: /service-b/\*\*  
 url: http://192.168.1.100:8080 # 静态路由到指定URL  
  
# Hystrix熔断配置  
hystrix:  
 command:  
 default:  
 execution:  
 isolation:  
 thread:  
 timeoutInMilliseconds: 30000 # 全局默认超时时间  
  
# Ribbon负载均衡配置  
ribbon:  
 ReadTimeout: 10000  
 ConnectTimeout: 5000  
 MaxAutoRetries: 1  
 MaxAutoRetriesNextServer: 2  
  
# Eureka服务发现配置  
eureka:  
 client:  
 service-url:  
 defaultZone: http://eureka-server:8761/eureka/ # 注册中心地址  
 instance:  
 prefer-ip-address: true  
  
# 日志配置  
logging:  
 level:  
 com.netflix.zuul: INFO  
   
# Actuator监控  
management:  
 endpoints:  
 web:  
 exposure:  
 include: "\*" # 开启所有端点，生产环境请按需开启