Gateway服务器

> 采用spring-cloud-gateway作为网关服务器。可以实现服务动态发现，负载均衡，访问认证，限流控制等功能。
>
> 网关采用【自动发现】+【自定义网关】组合实现，gateway路由匹配原理参考nginx（访问url在匹配到对应的路由后即停止）
>
> 网关匹配顺序需配置相应的order，自动发现路由order默认为0，自定义网关order根据需要调整

查看所有网关

http://{gateeayIp}:{gatewayPort}/actuator/gateway/routes

负载均衡

默认全局采用轮询规则 RoundRobinRule，可以根据需要调整

自定义ribbon规则参考：

custom ribbon config

```
#    ribbon
user-management:#
  ribbon:

#    NFLoadBalancerRuleClassName: com.netflix.loadbalancer.RandomRule #
#    NFLoadBalancerRuleClassName: com.netflix.loadbalancer.RoundRobinRule #

#    NFLoadBalancerRuleClassName: com.netflix.loadbalancer.RetryRule #
#    NFLoadBalancerRuleClassName: com.netflix.loadbalancer.
WeightedResponseTimeRule #
    NFLoadBalancerRuleClassName: com.netflix.loadbalancer.
BestAvailableRule #
    ConnectTimeout: 1000 #
    ReadTimeout: 5000 #
    OkToRetryOnAllOperations: true #
    MaxAutoRetriesNextServer: 3 #
    MaxAutoRetries: 2 #
```

访问控制

1.默认对所有请求进行拦截，统一调用单点登录服务API进行校验

2.不需要拦截的请求需进行过滤，配置如下：

权限 GlobalFilter

```
access:
  filters:
    - /forgetpsw
    - download
    - /user/logout
    - /user/login
    - /securityPolicys
    - /sysmessage
    - /ws/ReceiveOrderSendService.wsdl
```

限流

推荐采用gateway自带的令牌桶算法filter,配置如下:

限流

```
spring:
  application:
    name: gateway
  cloud:        # spring cloud gateway
    gateway:
      discovery:
        locator:
          enabled: true
          lowerCaseServiceId: true
      routes:
        - id: user-management
          uri: lb://user-management
          order: -1
          predicates:
            - Path=/user-management/forgetpsw/**
          filters:
            - StripPrefix=1
            - name: RequestRateLimiter
              args:
                key-resolver: "#{@ipKeyResolver}"
                redis-rate-limiter.replenishRate: 1
                redis-rate-limiter.burstCapacity: 1
```