**Springboot**

**Configuration Server**

To detect Configuration Server there are two methods

**1. Config First Bootstrap**

This the legacy form to detect the server but you don’t need to pass over Eureka Discovery Server which is in this case needed because we want to maintain Eureka URL with credentials in the Config Server for all Applications

**2. Discovery First Bootstrap**

In this Case you have to start Discovery Server to can configure your application, because the Config Server URL is stored in Eureka

Use first cae Config First Bootstrap

In pom.xml of service or in the BOM for all services

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-config</artifactId>  
</dependency>

<!-- To connect with config server on bootstrap -->  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-bootstrap</artifactId>  
</dependency>

in bootstrap.properties add, now you can start your server standalone without errors

# disable detect config server in local dev  
spring.cloud.config.enabled=false

If you want a cloud env (uses config server) so enable the config and add the server url

And in bootstrap-cloud.properties add

spring.cloud.config.uri=http://localhost:8088/  
#spring.cloud.bootstrap.enabled=true  
# enable detect config server  
spring.cloud.config.enabled=true

**Springboot 2.3.x**

**Spring cloud version Hoxton.R8 / R9**

1. **Use Eureka Discovery Server**

In bootstrap-cloud.properties

# allow eureka discovery (in this profile only)  
spring.cloud.discovery.enabled=true  
  
# set default zone in the properties file, without this you can't set it as  
# environment variable in docker compose EUREKA\_CLIENT\_SERVICEURL\_DEFAULTZONE  
eureka.client.serviceUrl.defaultZone=http://localhost:8761/eureka/  
eureka.instance.prefer-ip-address=true

set spring.cloud.discovery.serverId={config server app name}

**Tracing Sleuth-Zipkin**

**Springboot 2.3.x**

In BOM pom.xml add

<!-- spring-cloud-starter-zipkin in cloud Hoxton.SR8 -->  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-zipkin</artifactId>  
</dependency>

In Springboot 2.4.2

Add

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-sleuth</artifactId>  
</dependency>

And

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-sleuth-zipkin</artifactId>  
</dependency>

In every interested Project add in

application.prperties or application-tracing.prperties (profile tracing)

spring.zipkin.base-url=http://localhost:9411

Zipkin config

The configuration for zipkin can be maintained in the Config Server for all services

In config repo file “application-tracing.properties” we add

spring.zipkin.base-url=http://localhost:9411

in all services “application.properties” set

# default disable zipkin see tracing profile  
spring.zipkin.enabled=false

And in “application-tracing.properties” enable zipkin

spring.sleuth.sampler.probability=1  
spring.zipkin.enabled=true

**API Gateway:**

**Swagger UI:**

spring.cloud.gateway.routes[2].id=car-fleet-service  
spring.cloud.gateway.routes[2].uri=lb://car-fleet-service  
spring.cloud.gateway.routes[2].predicates[0]=Path=/cms/api/v1/cars, /cms/api/v1/cars/{segment}, /cms/swagger-ui/\*, /cms/v3/\*  
spring.cloud.gateway.routes[2].predicates[1]=Method=GET, POST, PUT, PATCH, DELETE  
spring.cloud.gateway.routes[2].filters[0]=RemoveRequestHeader=Cookie  
spring.cloud.gateway.routes[2].filters[1]=RewritePath=/cms/(?<segment>.\*), /$\{segment}

Important to add in path /cms/swagger-ui/\* and /cms/v3/\*

<http://localhost:8888/cms/swagger-ui/index.html>

Explore: /cms/v3/api-docs

RewritePath=/foo/(?<segment>.\*)**,** /$\{segment**}**

For a request path of /foo/bar, this will set the path to /bar before making the downstream request. Notice the $\ which is replaced with $ because of the YAML spec.

# swagger-ui custom path  
# when use path=swagger-ui.html using API Gateway URL jumps to server url  
# so don't use it with with Gateway or find a fix but you can still use  
# http://localhost:8888/cms/swagger-ui/index.html  
#springdoc.swagger-ui.path=/swagger-ui.html  
#springdoc.api-docs.path=/v3/api-docs