

【原创内容，转载、引用请注明出处】

1. 配置项解析

1.1 通用配置

```
# 应用名称，将会显示在Eureka界面的应用名称列
spring.application.name=config-service

# 应用端口，Eureka服务端默认为：8761
server.port=3333
```

1.2 eureka.server前缀的配置项

```
# 是否允许开启自我保护模式，缺省：true
# 当Eureka服务器在短时间内丢失过多客户端时，自我保护模式可使服务端不再删除失去连接的客户端
eureka.server.enable-self-preservation = false

# Peer节点更新间隔，单位：毫秒
eureka.server.peer-eureka-nodes-update-interval-ms =

# Eureka服务器清理无效节点的时间间隔，单位：毫秒，缺省：60000，即60秒
eureka.server.eviction-interval-timer-in-ms = 60000
```

1.3 eureka.instance前缀的配置项

```
# 服务名，默认取 spring.application.name 配置值，如果没有则为 unknown
eureka.instance.appname = eureka-client

# 实例ID
eureka.instance.instance-id = eureka-client-instance1

# 应用实例主机名
eureka.instance.hostname = localhost

# 客户端在注册时使用自己的IP而不是主机名，缺省：false
eureka.instance.prefer-ip-address = false

# 应用实例IP
eureka.instance.ip-address = 127.0.0.1

# 服务失效时间，失效的服务将被剔除。单位：秒，默认：90
eureka.instance.lease-expiration-duration-in-seconds = 90

# 服务续约（心跳）频率，单位：秒，缺省30
eureka.instance.lease-renewal-interval-in-seconds = 30

# 状态页面的URL，相对路径，默认使用 HTTP 访问，如需使用 HTTPS则要使用绝对路径配置，缺省：/info
```

```
eureka.instance.status-page-url-path = /info

# 健康检查页面的URL, 相对路径, 默认使用 HTTP 访问, 如需使用 HTTPS则要使用绝对路径配置, 缺省: /health
eureka.instance.health-check-url-path = /health
```

1.4 eureka.client前缀

```
# Eureka服务器的地址, 类型为HashMap, 缺省的Key为 defaultZone; 缺省的Value为
http://localhost:8761/eureka
# 如果服务注册中心为高可用集群时, 多个注册中心地址以逗号分隔。
eureka.client.service-
url.defaultZone=http://${eureka.instance.hostname}:${server.port}/eureka

# 是否向注册中心注册自己, 缺省: true
# 一般情况下, Eureka服务端是不需要再注册自己的
eureka.client.register-with-eureka = true

# 是否从Eureka获取注册信息, 缺省: true
# 一般情况下, Eureka服务端是不需要的
eureka.client.fetch-registry = true

# 客户端拉取服务注册信息间隔, 单位: 秒, 缺省: 30
eureka.client.registry-fetch-interval-seconds = 30

# 是否启用客户端健康检查
eureka.client.health-check.enabled = true

#
eureka.client.eureka-service-url-poll-interval-seconds = 60

# 连接Eureka服务器的超时时间, 单位: 秒, 缺省: 5
eureka.client.eureka-server-connect-timeout-seconds = 5

# 从Eureka服务器读取信息的超时时间, 单位: 秒, 缺省: 8
eureka.client.eureka-server-read-timeout-seconds = 8

# 获取实例时是否只保留状态为 UP 的实例, 缺省: true
eureka.client.filter-only-up-instances = true

# Eureka服务端连接空闲时的关闭时间, 单位: 秒, 缺省: 30
eureka.client.eureka-connection-idle-timeout-seconds = 30

# 从Eureka客户端到所有Eureka服务端的连接总数, 缺省: 200
eureka.client.eureka-server-total-connections = 200

# 从Eureka客户端到每个Eureka服务主机的连接总数, 缺省: 50
eureka.client.eureka-server-total-connections-per-host = 50
```

2. 配置项说明

Eureka中有一些配置项, 通过查看网上的资料可以大概知道其含义, 然而如果没有一个直观的指导, 并不能清晰地认识他们实际的作用效果。下面通过截图对部分配置项加以解释说明。

2.1 spring.application.name和eureka.instance.appname

同时配置时, `eureka.instance.appname` 优先级更高。

如果没有配置 `eureka.instance.appname` , 则使用 `spring.application.name` 的值, 如果连 `spring.application.name` 都没有配置, 则取**unknown**。

该配置项对应Eureka界面下图红框的内容:

Instances currently registered with Eureka			
Application	AMIs	Availability Zones	Status
EUREKA-TEST	n/a (1)	(1)	UP (1) - localhost:eureka-server:8080

2.2 eureka.instance.instance-id

配置项 `eureka.instance.instance-id` 的值决定了下图右侧红框中的显示内容:

Instances currently registered with Eureka			
Application	AMIs	Availability Zones	Status
EUREKA-SERVER	n/a (1)	(1)	UP (1) - eureka-client-instance1

如果没有设置 `eureka.instance.instance-id` , 那么显示的值将是一个由Eureka自动判断生成的编号:

Instances currently registered with Eureka			
Application	AMIs	Availability Zones	Status
EUREKA-SERVER	n/a (1)	(1)	UP (1) - localhost:eureka-server:8080

2.3 eureka.instance.prefer-ip-address、eureka.instance.hostname、eureka.instance.ip-address

`eureka.instance`前缀的配置项中, 这几个配置项绝对烧脑, 根据[周立的博客](#), 在

`eureka.instance.prefer-ip-address = true` 时, 实例优先使用

`eureka.instance.ip-address` 的值进行注册, 如果没有配置 `eureka.instance.ip-address` , 则使用第一个非回环IP地址进行注册。

此时, 我们打开Eureka的界面, 在实例上右键, 复制链接地址; 或将鼠标放在下图右上方的链接上 (不点击), 就可以获取实例的地址, 如下图左下角, 可见此时实例注册的是IP:

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
UNKNOWN	n/a (1)	(1)	UP (1) - localhost:8080

General Info

Name	Value
total-avail-memory	397mb
environment	test
num-of-cpus	4
current-memory-usage	282mb (71%)
server-uptime	00:00
registered-replicas	http://localhost:8080/eureka/
unavailable-replicas	http://localhost:8080/eureka/
available-replicas	

Instance Info

192.168.56.1:8080/actuator/info

而当 `eureka.instance.prefer-ip-address = false` 时，同样的方式可以查看实例注册地址采用了主机名 `eureka.instance.hostname` 的值：

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
EUREKA-SERVER	n/a (1)	(1)	UP (1) - eureka-client-instance1

General Info

Name	Value
total-avail-memory	412mb
environment	test
num-of-cpus	4
current-memory-usage	215mb (52%)
server-uptime	00:00
registered-replicas	
unavailable-replicas	
available-replicas	

Instance Info

Name	Value
localhost:8080/actuator/info	192.168.56.1

3. 配置Bean源码

最后，对应于本文出现的配置项，Eureka中定义的源码类如下。

eureka.server前缀的配置项

```
org.springframework.cloud.netflix.eureka.server.EurekaServerConfigBean
```

eureka.instance前缀的配置项

```
org.springframework.cloud.netflix.eureka.EurekaInstanceConfigBean
```

eureka.client前缀的配置项

```
org.springframework.cloud.netflix.eureka.EurekaClientConfigBean
```

参考博文:

[周立的博客](#)

<https://www.cnblogs.com/liaojie970/p/8807023.html>

<https://www.cnblogs.com/li3807/p/7282492.html>

<https://www.jianshu.com/p/c18d140ad9f6>