docker 安装并启动graylog

# 前言我的是mac本安装的

## 1、创建目录 /usr/local/graylog

## 2、新建文件 graylog.yml

### 1、命令 healerjean$ sudo vim graylog.yml

### 2、文件中写入

version: '2'

services:

mongo:

image: "mongo:3"

elasticsearch:

image: "elasticsearch:2"

command: "elasticsearch -Des.cluster.name='graylog'"

graylog:

image: graylog2/server:2.2.1-1

environment:

GRAYLOG\_PASSWORD\_SECRET: somepasswordpepper

GRAYLOG\_ROOT\_PASSWORD\_SHA2: 8c6976e5b5410415bde908bd4dee15dfb167a9c873fc4bb8a81f6f2ab448a918

GRAYLOG\_WEB\_ENDPOINT\_URI: http://127.0.0.1:9000/api

depends\_on:

- mongo

- elasticsearch

ports:

- "9000:9000"

- "12201:12201"

- "12202:12202"

- "514:514"

~

version: '2'

services:

*# MongoDB: https://hub.docker.com/\_/mongo/*

mongodb:

image: mongo:3

*# Elasticsearch: https://www.elastic.co/guide/en/elasticsearch/reference/5.6/docker.html*

elasticsearch:

image: docker**.**elastic**.**co**/**elasticsearch**/**elasticsearch:5.6**.**3

environment:

**-** http**.**host**=**0.0**.**0.0

**-** transport**.**host**=**localhost

**-** network**.**host**=**0.0**.**0.0

*# Disable X-Pack security: https://www.elastic.co/guide/en/elasticsearch/reference/5.6/security-settings.html#general-security-settings*

**-** xpack**.**security**.**enabled**=**false

**-** "ES\_JAVA\_OPTS=-Xms512m -Xmx512m"

ulimits:

memlock:

soft: **-**1

hard: **-**1

mem\_limit: 1g

*# Graylog: https://hub.docker.com/r/graylog/graylog/*

graylog:

image: graylog**/**graylog:2.4**.**0**-**1

environment:

*# CHANGE ME!*

**-** GRAYLOG\_PASSWORD\_SECRET**=**somepasswordpepper

*# Password: admin*

**-** GRAYLOG\_ROOT\_PASSWORD\_SHA2**=**8c6976e5b5410415bde908bd4dee15dfb167a9c873fc4bb8a81f6f2ab448a918

**-** GRAYLOG\_WEB\_ENDPOINT\_URI**=**http:**//**127.0**.**0.1:9000**/**api

links:

**-** mongodb:mongo

**-** elasticsearch

depends\_on:

**-** mongodb

**-** elasticsearch

ports:

*# Graylog web interface and REST API*

**-** 9000:9000

*# Syslog TCP*

**-** 514:514

*# Syslog UDP*

**-** 514:514**/**udp

*# GELF TCP*

**-** 12201:12201

*# GELF UDP*

**-** 12201:12201**/**udp

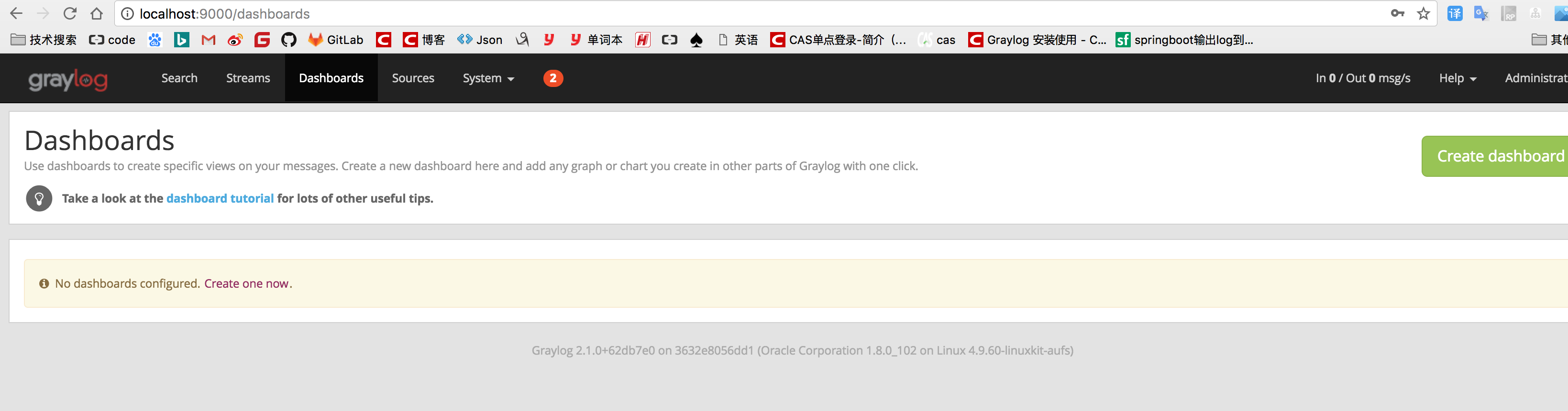
## 3、启动(使用docker compose部署服务)

### healerjean$ sudo docker-compose -f graylog.yml up -d

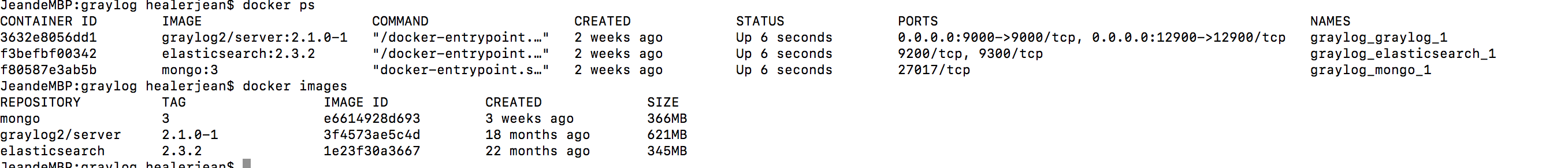
**或者文件名为**docker-compose.yml

**直接输入命令 docker-compose up -d （d后台运行）**

## 4、浏览器中访问 <http://localhost:9000/> （admin/admin）

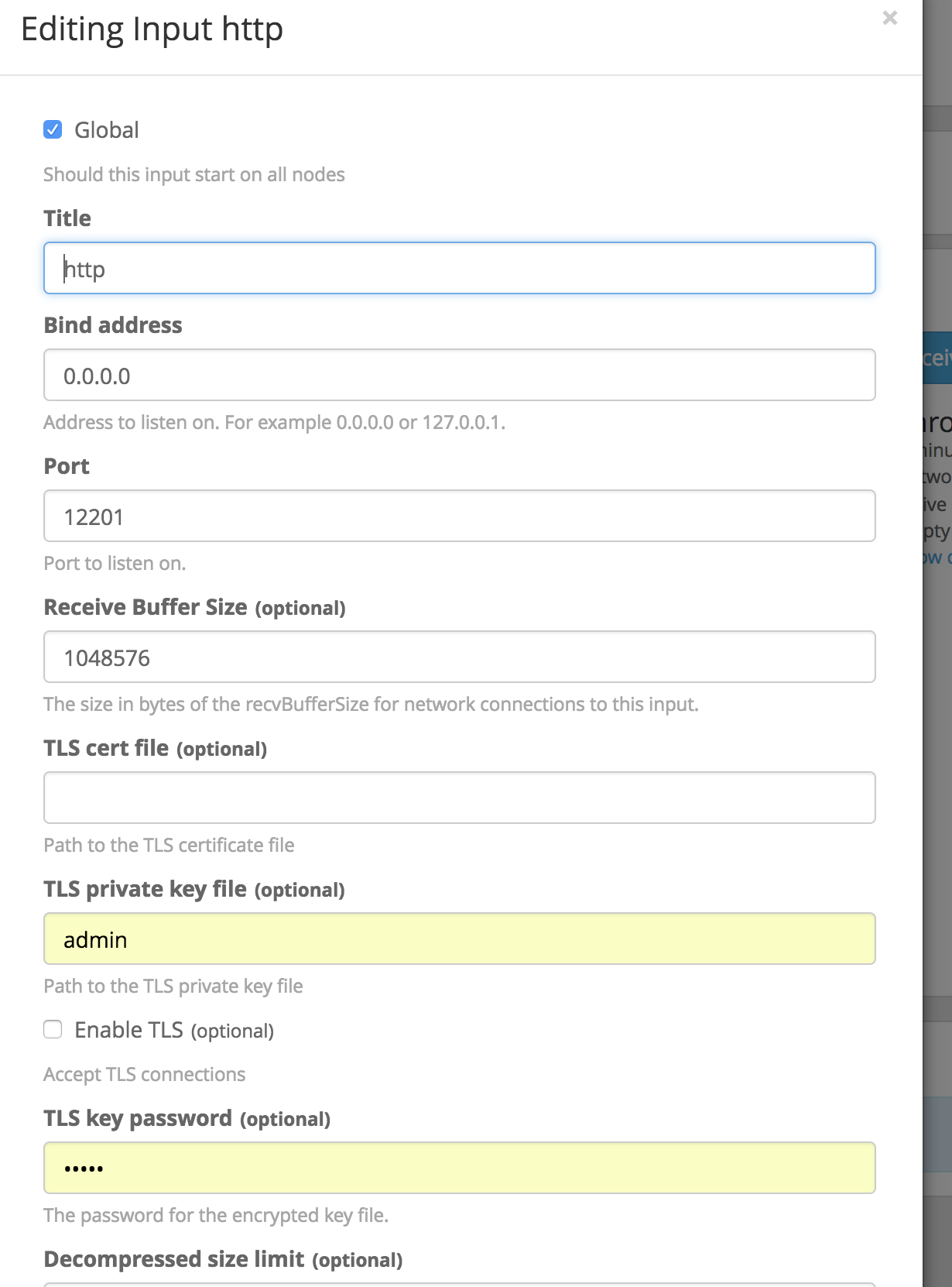


## 5、观察镜像



## 6、测试 http upd

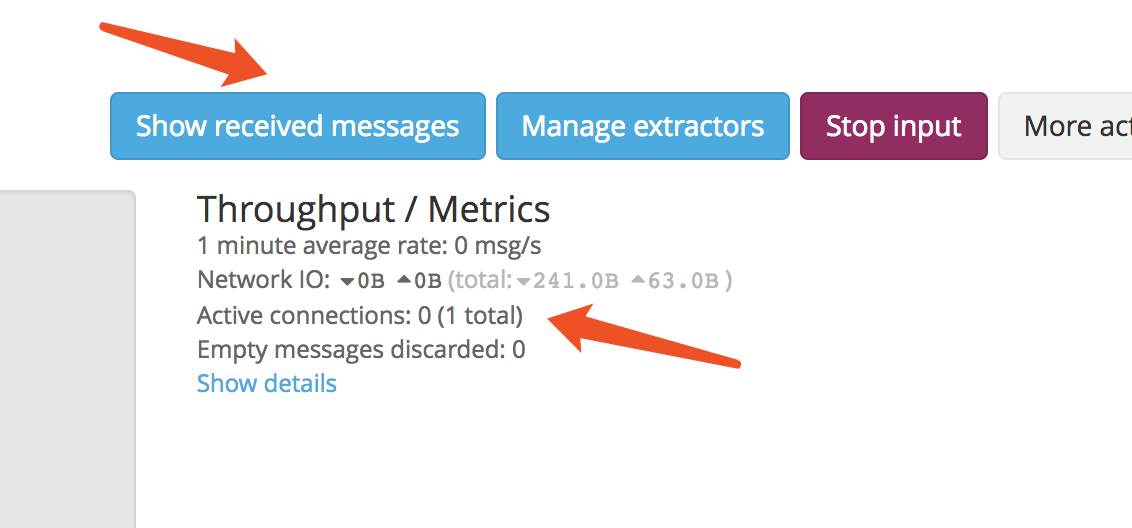
### 1、新建input GELF HTTP

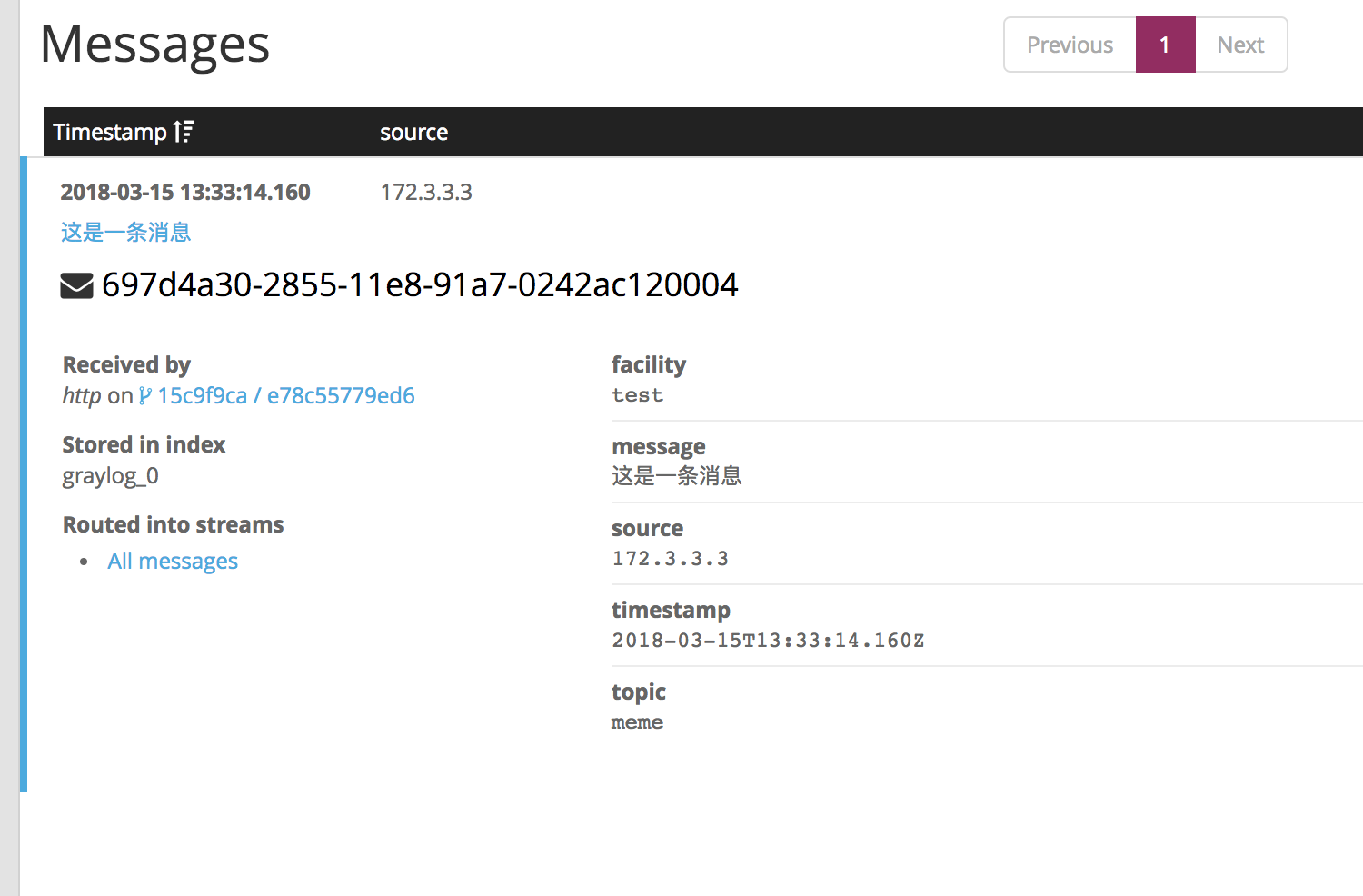


### 2、终端中可以输入下面的进行测试

#### curl -XPOST http://localhost:12201/gelf -p0 -d '{"message":"hello这是一条消息", "host":"127.0.0.1", "facility":"test", "topic": "meme"}'

#### 开始查看

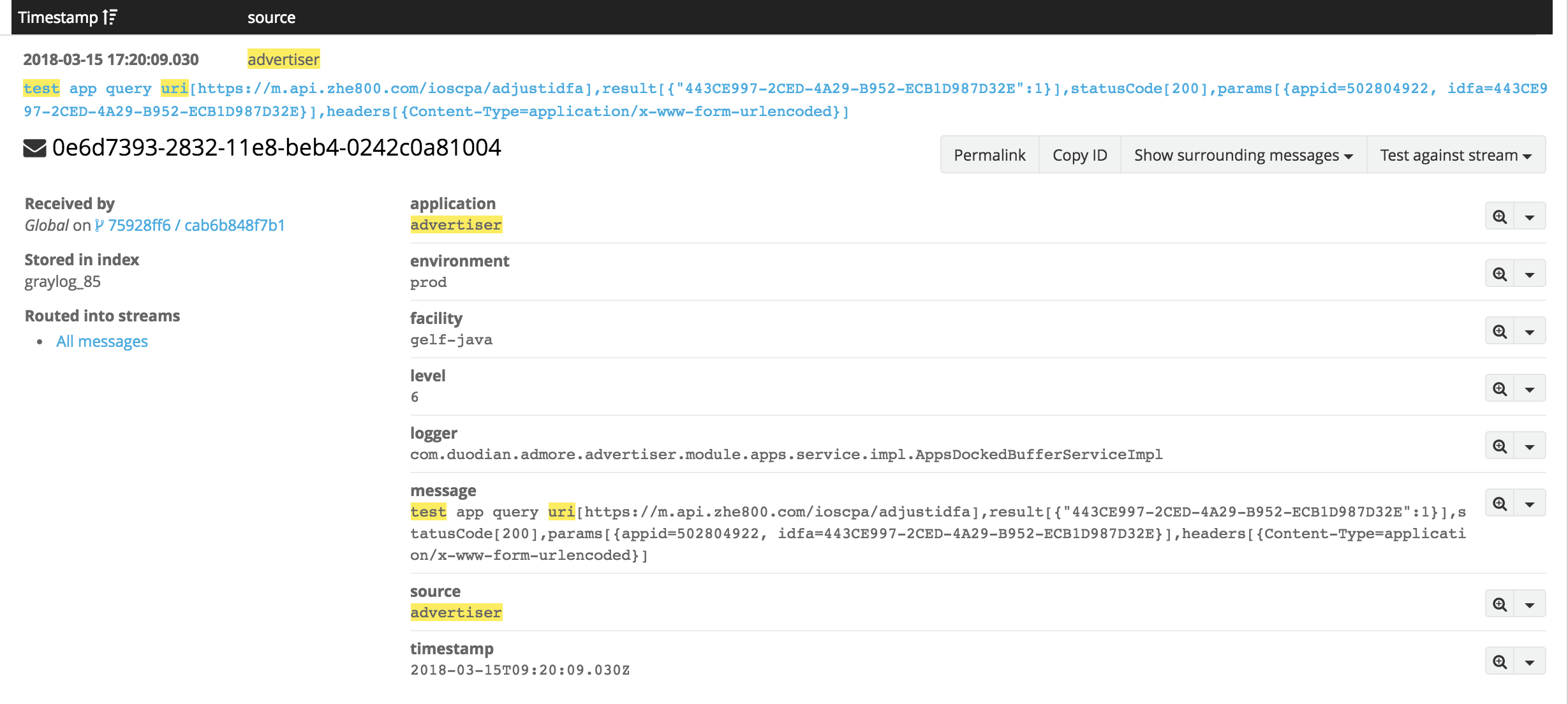




# 二、springBoot集成

## 1、对应日志内容，不过别忘记打日志哟

*<?*xml version="1.0" encoding="UTF-8"*?>*<configuration>  
  
 <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">  
 <encoder>  
 <pattern>%d{HH:mm:ss.SSS} %-5level %logger{36} - %msg%n</pattern>  
 </encoder>  
 </appender>  
  
  
 <appender name="logging" class="com.github.pukkaone.gelf.logback.GelfAppender">  
 <graylogHost>127.0.0.1</graylogHost>  
 <originHost>advertiser</originHost> <!--source,来源-->  
 <levelIncluded>true</levelIncluded>  
 <locationIncluded>false</locationIncluded>  
 <loggerIncluded>true</loggerIncluded>  
 <markerIncluded>false</markerIncluded>  
 <mdcIncluded>false</mdcIncluded>  
 <threadIncluded>false</threadIncluded>  
 <facility>gelf-java</facility> <!--facility，随便写呗-->  
 <additionalField>application=advertiser</additionalField> <!--application，应用-->  
 <additionalField>environment=prod</additionalField> <!--environment，环境-->  
 </appender>  
  
  
 <root level="info">  
 <appender-ref ref="STDOUT" />  
 <appender-ref ref="logging" />  
 </root>  
</configuration>



## 2、查询语法 (and后面可以加message内容，source就是source的值,或者也可以链接其他选项)

### source:advertiser AND test AND click

### source:callback AND EED2EE41-7EDA-4C89-9AC0-B95B226F44E1 看回调的命令