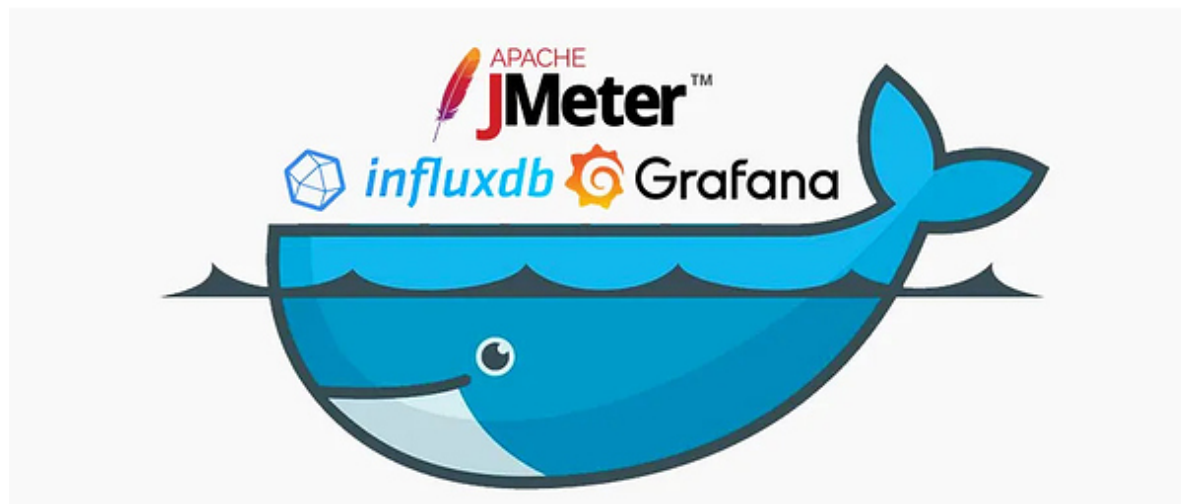


Docker+JMeter+InfluxDB+Grafana 搭建性能监控平台

1 简介



JMeter自带有原生报告，为什么要搭建性能监控平台？

聚合报告													
名称: 聚合报告													
注释:													
所有数据写入一个文件													
文件名										浏览...	显示日志内容: <input type="checkbox"/> 仅错误日志 <input type="checkbox"/> 仅成功日志		配置
Label	# 样本	平均值	中位数	90% 百...	95% 百...	99% 百...	最小值	最大值	异常 %	吞吐量	接收 KB/...	发送 KB/...	
HTTP请求	2767	7	6	11	15	18	4	83	0.07%	136.7/sec	333.23	15.74	
总体	2767	7	6	11	15	18	4	83	0.07%	136.7/sec	333.23	15.74	

JMeter原生报告的缺点:

- 无法实时共享
- 报告信息的展示不美观

需求方案

为了解决上述问题，可以通过 InfluxDB + Grafana解决：

- **InfluxDB**：是一个**开源分布式指标数据库**，使用 Go 语言编写，无需外部依赖
应用：**性能监控**，**应用程序指标**，**物联网传感器数据**和**实时分析**等的后端存储
- **Grafana**：Grafana是一款用Go语言开发的**开源数据可视化工具**，可以做数据监控和数据统计，用于**将存储于InfluxDB中的数据以图表的形式展示出来**

应用：**将 Jmeter 的数据导入 InfluxDB，再用 Grafana 从 InfluxDB 中获取数据并以特定的模板进行展示**

2 搭建性能平台

2.1 部署InfluxDB

部署方法：Docker部署 提前安装docker

Docker常见基础命令

列出已下载docker镜像： `docker images`

列出运行中的docker 容器： `docker ps`

列出运行中+已停止docker 容器： `docker ps -a`

停止容器： `docker stop [docker container id]`

启动容器： `docker start [docker container id]`

删除容器： `docker rm [docker container id]`

1、下载InfluxDB的镜像，默认为下载最新的镜像：

```
$ docker pull influxdb
```

2、启动一个容器，并将端口 8083 和 8086 映射出来

```
$ docker run -d --name jmeter-influx -p 8083:8083 -p 8086:8086 influxdb
```

各个参数含义：

-d：容器在后台运行

-p：将容器内端口映射到宿主机端口，格式为 宿主机端口:容器内端口；8083是influxdb的web管理工具 端口，8086是influxdb的HTTP API端口

--name：容器名称

最后是镜像名称

检查是否启动成功： `docker images`

```
root@zx /] # docker run -d --name jmeter-influx -p 8083:8083 -p 8086:8086 influxdb
3bd972ee417379188e24b1065c656dbcf616f0812cafec0863b650ed829b0542
```

```
root@zx /] # docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
grafana/grafana	latest	651ff2dc930f	6 months ago	187MB
influxdb	latest	f3a5428c2469	6 months ago	307MB
registry.cn-beijing.aliyuncs.com/ericzx/django_proj	v1.0-event-weather-etc	69f104528ca5	13 months ago	1.13GB
nello-world	latest	fce289e99eb9	2 years ago	1.84kB

3、进入容器内部，创建名为jmeter的数据库：

进入 jmeter-influx 容器

```
$ docker exec -it jmeter-influx bash
```

4、启动控制台客户端，进入 influx

5、在容器内部创建jmeter数据库，用于收集JMeter发送的压测数据

6、使用JMeter 库，还没有配置jmeter， select 查看数据,这个时候应该还是没有数据的：

图例是已配置好jmeter

```
$ > use jmeterUES
```

```
Using database jmeter
```

```
> select * from jmeter
```

```
>
```

```
root@9bd972ee4173: /# influx
Connected to http://localhost:8086 version 1.8.3
InfluxDB shell version: 1.8.3
> show databases
name: databases
name
----
internal
jmeter
> create database JMeter
> show databases
name: databases
name
----
internal
jmeter
JMeter
> use jmeter
Using database jmeter
> select * from jmeter
name: jmeter
time                application avg    responseCode    count countError endedT hit    max    maxAT    meanAT    min    minAT    pct90.0    pct95.0
      pct99.0                rb    started status transaction
-----
1622385547941000000 lagou          0          internal          1          0          1          39          39          39          39
1622385547985000000 lagou          39          all          HTTP请求          1          0          1          39          39          39          39
```

1、进入数据库

2、查看当前有几个库

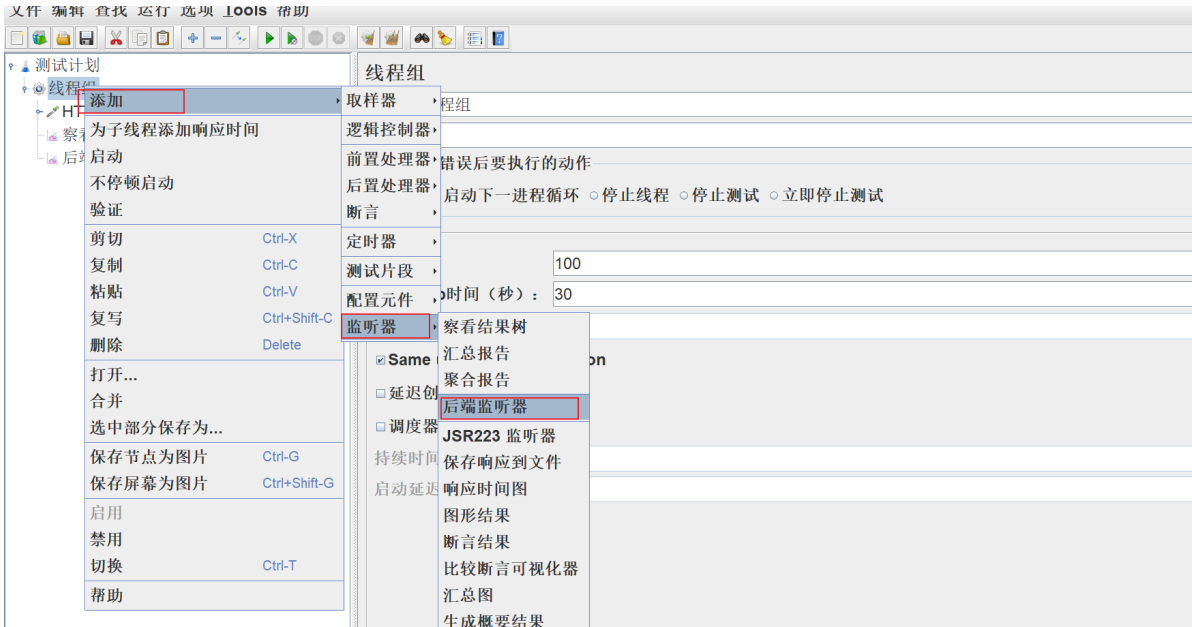
3、创建jmeter库

4、再次查看 存在创建jmeter 常见成功

5、切换到jmeter库，查询连接数据

2.2 配置JMeter

1、将 jmeter 的测试数据导入 influxDB , jmeter添加后端监听器



配置好的后端监听器：

后端监听器	
名称：	后端监听器
注释：	
后端监听器实现	org.apache.jmeter.visualizers.backend.influxdb.InfluxdbBackendListenerClient
异步队列大小	5000
参数	
名称：	值
influxdbMetricsSender	org.apache.jmeter.visualizers.backend.influxdb.HttpMetricsSender
influxdbUrl	http://192.168.85.128:8086/write?db=jmeter
application	lagou
measurement	jmeter
summaryOnly	false
samplersRegex	*
percentiles	90;95;99
testTitle	Test name
eventTags	

2、主要配置说明

后端监听器 选择 influxdb 所对应的：

后端监听器	
名称：	后端监听器
注释：	
后端监听器实现	org.apache.jmeter.visualizers.backend.influxdb.InfluxdbBackendListenerClient
异步队列大小	5000
参数	
名称：	值
influxdbMetricsSender	org.apache.jmeter.visualizers.backend.influxdb.HttpMetricsSender
influxdbUrl	http://192.168.85.128:8086/write?db=jmeter
application	lagou
measurement	jmeter
summaryOnly	false
samplersRegex	*
percentiles	90;95;99
testTitle	Test name
eventTags	

- **influxdbUrl**：需要改为自己 influxdb 的部署 ip 和映射端口，我这里是部署在centos，端口是容器启动时映射的** 8086 端口，db 后面跟的是刚才创建的数据库名称
- **application**：可根据需要自由定义，只是注意后面在 grafana 中选对即可
- **measurement**：表名，默认是 jmeter，也可以自定义
- **summaryOnly**：选择 true 的话就只有总体的数据，false 会将每个 transaction 都分别记录

3、运行验证

运行 Jmeter 脚本，然后再次在 influxdb 中查看数据，发现类似下面的数据说明输入导入成功：

```
> use jmeter
Using database jmeter
> select * from jmeter
name: jmeter
time           application avg      count countError endedT hit    max    maxAT meanAT min minAT pct90.0      pct95.0
           pct99.0          rb      responseCode      transaction
           -----
1622385547941000000 lagou      0      internal
39                      39      1
1622385547985000000 lagou      2497    HTTP请求
118                      1      0      1      39      39      39
1622385547985000000 lagou      2497    all
118                      1      0      1      39      39      39
1622385547986000000 lagou      39      all
118                      1      0      1      39      39      39
1622385547986000000 lagou      39      ok      HTTP请求
1622385547986000000 lagou      1      internal
1622385589083000000 lagou      0      0      0      0
```

2.3部署Grafana

下载grafana的镜像

Grafana部署

1、下载grafana的镜像：

```
$ docker pull grafana/grafana
```

```
[root@zx /] # docker pull grafana/grafana
Using default tag: latest
latest: Pulling from grafana/grafana
339de151aab4: Pull complete
54c6e53234ff: Pull complete
cb4d5838b0bb: Pull complete
1bdf9322a25e: Pull complete
3363e2a9eb97: Pull complete
4f4fb700ef54: Pull complete
07f050e4d33d: Pull complete
787e6b8012a9: Pull complete
Digest: sha256: 4e5835bcfd55cf72563a06932f10c75d9d92a0e1334a4c83eaa9c5b897370b25
Status: Downloaded newer image for grafana/grafana:latest
docker.io/grafana/grafana:latest
```

2、启动一个grafana容器,将3000端口映射出来：

端口映射是指将容器内部的端口映射到主机上的一个端口，使得主机可以通过该端口访问容器内的服务。

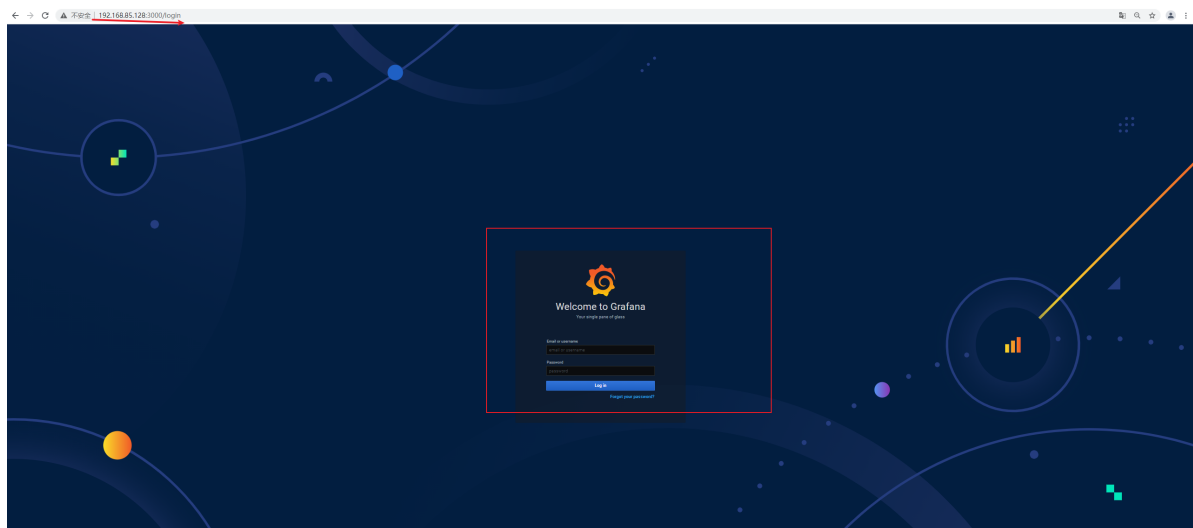
```
$ docker run -d --name grafana -p 3000:3000 grafana/grafana
```

```
[root@zx /] # docker run -d --name grafana -p 3000:3000 grafana/grafana
c71899db885c142d9d49068c79fedcf942c83f34e7226673c017a79cb75ba316
```

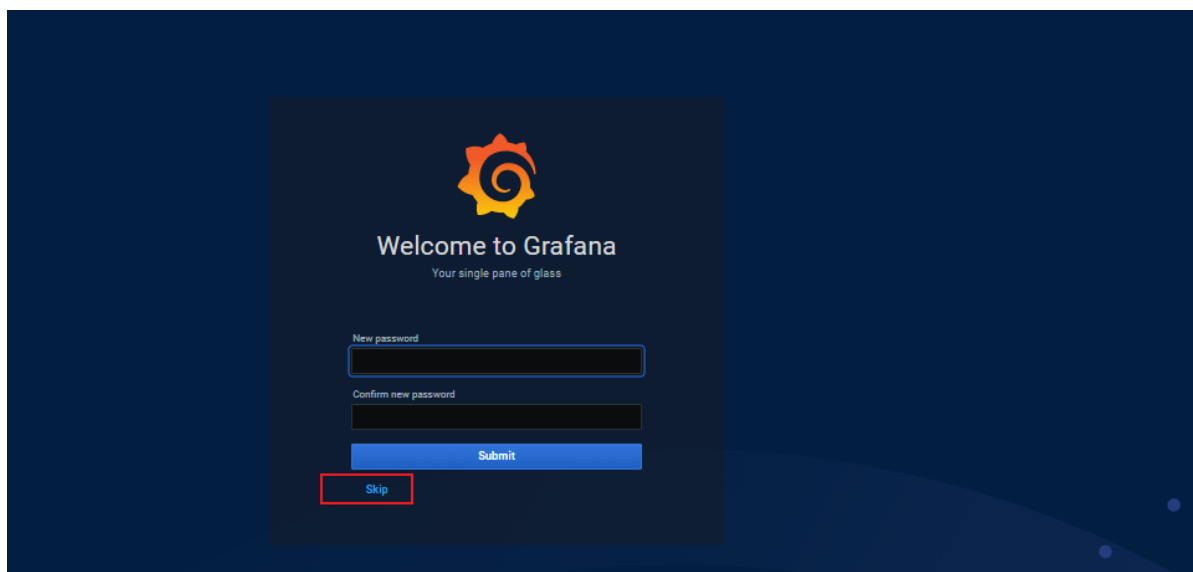
3、网页端访问ip:3000验证部署成功

```
[root@zx /]# ifconfig
docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    inet6 fe80::42:72ff:fe7b:864e prefixlen 64 scopeid 0x20<link>
    ether 02:42:72:7b:86:4e txqueuelen 0 (Ethernet)
    RX packets 59 bytes 4304 (4.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 78 bytes 20413 (19.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

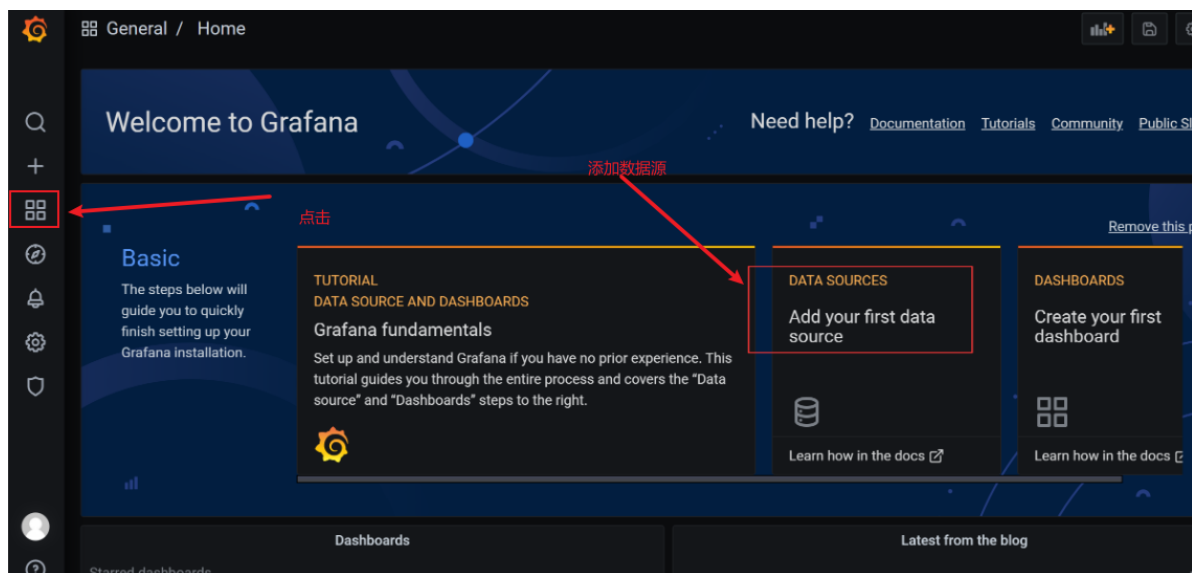
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.85.128 netmask 255.255.255.0 broadcast 192.168.85.255
    inet6 fe80::38be:2bfa:90c4:5b75 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:5a:27:66 txqueuelen 1000 (Ethernet)
    RX packets 370011 bytes 548746674 (523.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 45777 bytes 3476543 (3.3 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```



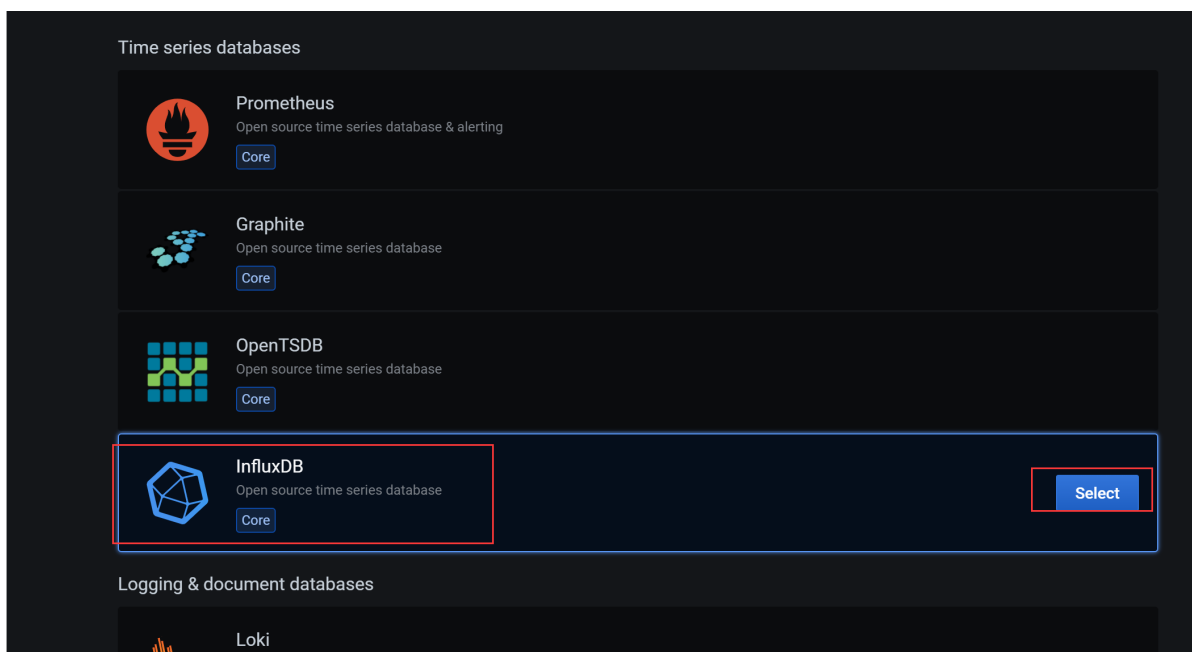
用户名: admin 密码: admin 输入一遍后 点击skip



4、选择添加数据源



5、找到并选择 influxdb :



6、配置数据源

数据源设置

Name 设置数据源名称

URL InfluxDB的访问地址

Access 选择Browser

Database InfluxDB中创建的数据库名

Settings

Name

InfluxDB-lagou

Default

Query Language

InfluxQL

HTTP

URL

http://192.168.85.128:8086

Access

Browser

Help

Auth

Basic auth

With Credentials

InfluxDB Details

Database Access

Setting the database for this datasource does not deny access to other databases. The InfluxDB query syntax allows switching the database in the query. For example: `SHOW MEASUREMENTS ON _internal` or `SELECT * FROM "_internal".."database" LIMIT 10`

To support data isolation and security, make sure appropriate permissions are configured in InfluxDB.

Database

JMeter

User

Password

Password

HTTP Method

Choose

Min time interval

10s

Max series

1000

✓

Data source is working

Save & Test

Delete

Back

数据源创建成功时会有绿色的提示：

Database

jmeter

User

Password

configured

reset

HTTP Method

GET

✓

Datasource updated

Database Access

Setting the database for this datasource does not deny access to other databases. The InfluxDB query syntax allows switching the database in the query. For example: `SHOW MEASUREMENTS ON _internal` or `SELECT * FROM "_internal".."database" LIMIT 10`

To support data isolation and security, make sure appropriate permissions are configured in InfluxDB.

Min time interval

10s

✓

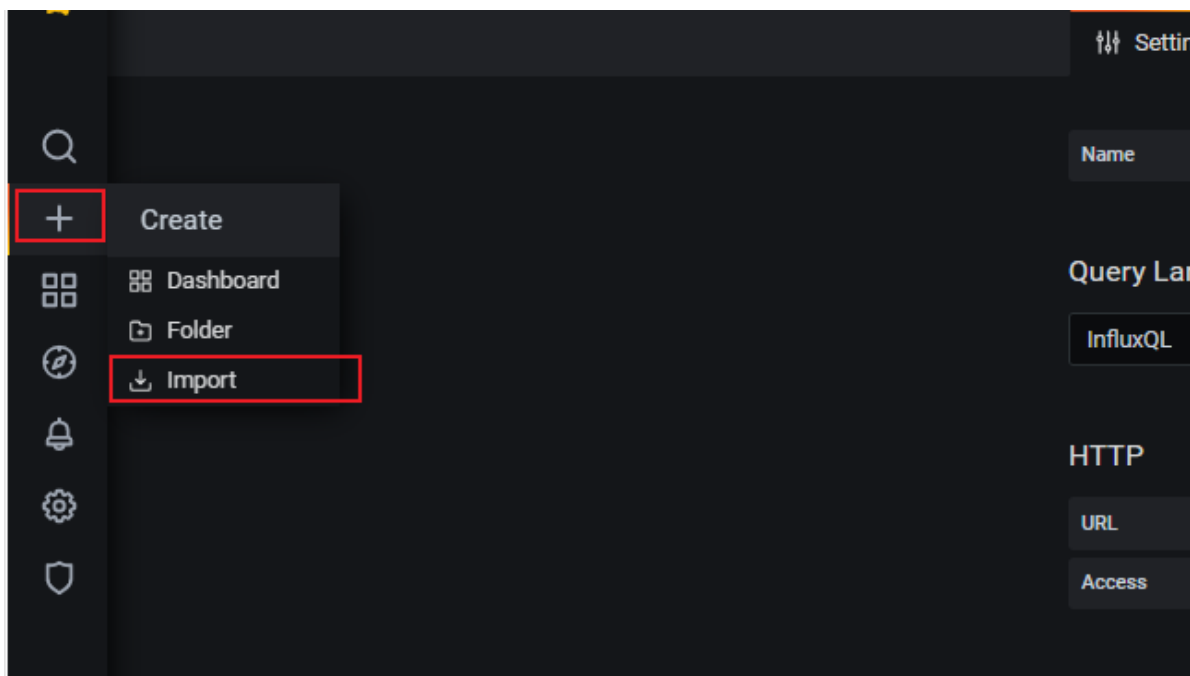
Data source is working

Save & Test

Delete

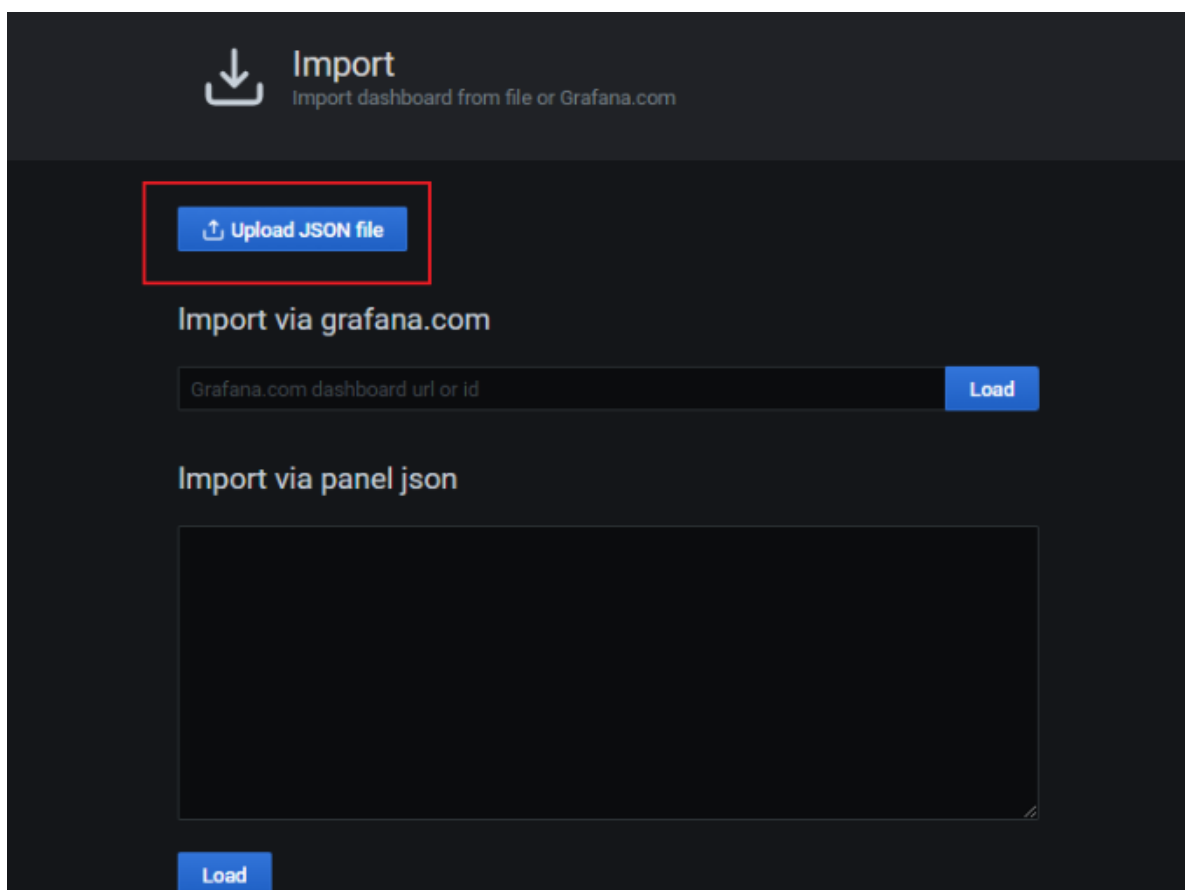
Back

7、导入模板



模板导入分别有以下3种方式：

- 直接输入模板id号
- 直接上传模板json文件
- 直接输入模板json内容




8、下载模板

地址：<https://grafana.com/grafana/dashboards/5496>

Grafana Labs Grafana Products Open Source Learn Downloads **Contact us** Login

All dashboards » Apache JMeter Dashboard using Core InfluxdbBackendListenerClient



Apache JMeter Dashboard
by Philippe M.
Monitor your Apache JMeter load test in real time with InfluxDB and Grafana. Get overall summary, errors details and particular transaction response times.
Last updated: 3 years ago
Start with Grafana Cloud and the new FREE tier. Includes 10K series Prometheus or Graphite Metrics and 50gb Loki Logs

Downloads: 12424
Reviews: 5
★ ★ ★ ★ ★
Add your review!

Overview **Revisions** Reviews

Dashboard Revisions

Revision	Description	Created	
1	Monitor your Apache JMeter load test in real time with InfluxDB and Grafana.	April 10th 2018, 10:21 pm	Download

Get this dashboard:
5496
Copy ID to Clipboard

Download JSON
How do I import this dashboard?

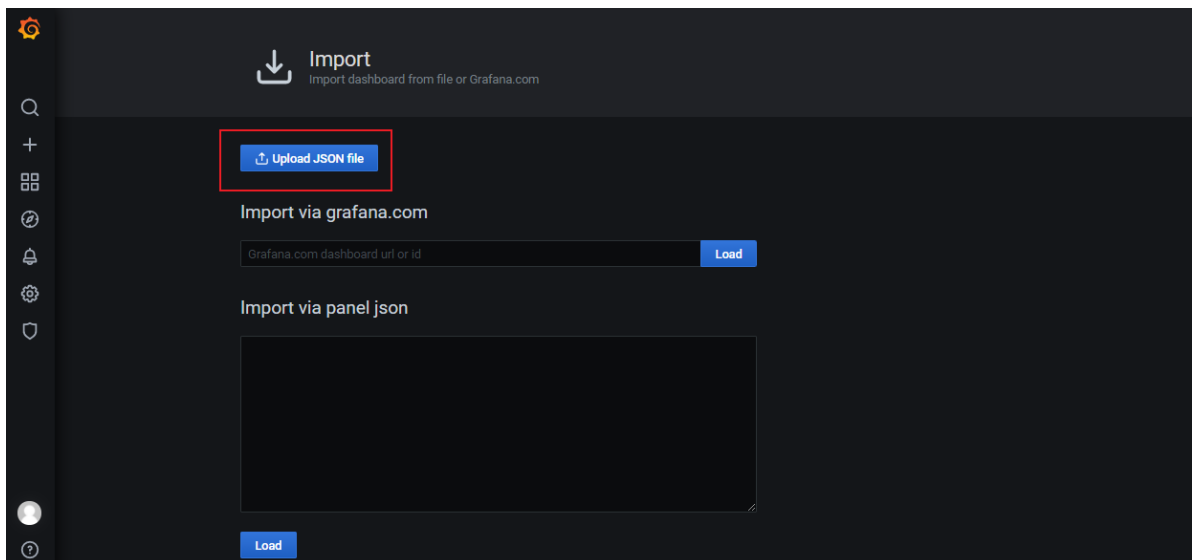
Dependencies:

- GRAFANA 4.6.2
- GRAPH
- INFLUXDB 1.4.0
- SINGLESTAT

9、依次点击“+”——>“Import”——>“Upload JSON file”按钮，选中刚下载的JSON文件

（也可以将“<https://grafana.com/grafana/dashboards/5496>”复制到“Import via grafana.com”中，点击“Load”按钮）

（还可以在“Import via panel json中直接复制json”）



模板设置

Name 设置模板名称

DB name 选择“InfluxDB”

Measurement name （待说明）

Backend send interval 设置每5秒刷新一次

Options

Name
Apache JMeter Dashboard using Core InfluxdbBackendListenerClient 自定义

Folder
General

Unique identifier (uid)
The unique identifier (uid) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The uid allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.
Change uid

DB name
InfluxDB-lagou

Measurement name
jmeter

Backend send interval
5

Import Cancel

9、选择创建好的application

General / Apache JMeter Dashboard using Core InfluxdbBackendListenerClient ☆ 🔗

data_source InfluxDB-lagou application lagou transaction HTTP请求 Start/stop marker ☒

Summary

Total Requests 0 Requests	Failed Requests 0 Failed	Received Bytes 0	Sent Bytes 0
Total Throughput		Total Errors	

注意： 如果我们修改过表名，也就是在jmeter的Backend Listener的measurement配置(默认为jmeter)，这个时候就需要去设置中进行修改(我这里使用的就是默认的，所以无需修改)：

Apache JMeter Dashboard using Core InfluxdbBackendListenerClient / Settings

General Annotations Variables Links Versions Permissions JSON Model

Save dashboard Save As...

Variables

Show dependencies New

Variable	Definition					
data_source	influxdb	✓	🔗	+	📄	🗑️
application	SHOW TAG VALUES FROM "JMeter" WITH KEY = "application"	✓	🔗	↑	+	📄
transaction	SHOW TAG VALUES FROM "JMeter" WITH KEY = "transaction" WHERE "application" =~ /"\$application\$"/ ...	✓	🔗	↑	+	📄
measurement_name	JMeter	✓	🔗	+	+	📄
send_interval	5	✓	🔗	+		📄

Unknown Variables ⓘ

Variable aggregation request

General / Apache JMeter Dashboard using Core InfluxdbBackendListenerClient ☆ 🔊

Last 5 minutes 🔍 5s ↻ 📄 ⚙️

data_source InfluxDB-lagou application lagou transaction HTTP请求 Start/stop marker ☒

Summary

Total Requests	Failed Requests	Received Bytes	Sent Bytes	Error Rate %
0 Requests	0 Failed	0	0	0

5s刷新

Off 5s 10s 30s 1m 5m 15m 30m 1h 2h 1d

Total Throughput Total Errors Active Threads

General / Apache JMeter Dashboard using Core InfluxdbBackendListenerClient ☆ 🔊

Last 5 minutes 🔍 5s ↻ 📄 ⚙️

Total Requests	Failed Requests	Received Bytes	Sent Bytes	Error Rate %
26295 Requests	6134 Failed	60 MiB	2 MiB	23.33%

Total Throughput Total Errors Active Threads

Req / s min max avg Num of Errors total Threads current

30.00 2.96 K 739.87 6 K

