

# 指标管理系统测试环境使用说明

肖燊 – yu.xiao@seaboxdata.com – Version 1.0

---

## Table of Contents

测试环境要求

测试环境的内容

组件

数据

测试环境设置步骤

初次启动测试环境

关闭测试环境

重新启动测试环境

---

## 测试环境要求

- 一台Linux虚拟机（CentOS或Ubuntu均可）。虚拟机上如果已经安装MySQL，需要将其关闭。

## 测试环境的内容

### 组件

- Hadoop 2.9.2
- Hive 2.3.7
- MySQL 5.7

### 数据

- 指标数据源测试数据，在Hive的metric\_src\_db数据库中，可以通过以下链接访问：
  - jdbc:hive2://localhost:10000/metric\_src\_db
- 指标管理系统测试数据，在MySQL的metric\_db数据库中，可以通过以下链接访问：
  - jdbc:mysql://localhost:3306/metric\_db?user=root&password=root&characterEncoding=utf8
- 指标仓库测试数据，在MySQL的metric\_warehouse数据库中，可以通过以下链接访问：
  - jdbc:mysql://localhost:3306/metric\_warehouse?user=root&password=root&characterEncoding=utf8

## 测试环境设置步骤

1. 在虚拟机上安装Docker和Docker Compose（参考官方文档）。
2. 打开链接（<https://pan.baidu.com/s/11569C65nuNvDww06yHI0ng>，提取码：9fey），进入test目录下载下列文件
  - metric-sys-test.tar.gz
  - metric-sys-test-images.tar.gz并将这两个文件上传到虚拟机。
3. 执行以下命令将metric-sys-test-images.tar.gz导入docker，

```
$ gzip -d metric-sys-test-images.tar.gz
$ docker load -i metric-sys-test-images.tar
```

#### 4. 在虚拟机任意目录下展开metric-sys-test.tar.gz

```
$ tar xvzf metric-sys-test.tar.gz
```

## 初次启动测试环境

进入metric-sys-test目录，执行：

```
$ docker-compose up
```

等待一会儿，直到观察到端口10000被占用：

```
$ netstat -tlnp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:25           0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:50010          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:50075          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:50020          0.0.0.0:*                LISTEN      -
tcp        0      0 127.0.0.1:9000         0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:50090          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:2222           0.0.0.0:*                LISTEN      -
tcp        0      0 127.0.0.1:32975        0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:10000          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:10002          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:50070          0.0.0.0:*                LISTEN      -
tcp        0      0 0.0.0.0:22             0.0.0.0:*                LISTEN      -
tcp6       0      0 :::8088                :::*                    LISTEN      -
tcp6       0      0 :::1:25                :::*                    LISTEN      -
tcp6       0      0 :::13562                :::*                    LISTEN      -
tcp6       0      0 :::8030                 :::*                    LISTEN      -
tcp6       0      0 :::8031                 :::*                    LISTEN      -
tcp6       0      0 :::8032                 :::*                    LISTEN      -
tcp6       0      0 :::8033                 :::*                    LISTEN      -
tcp6       0      0 :::8040                 :::*                    LISTEN      -
tcp6       0      0 :::40296                :::*                    LISTEN      -
tcp6       0      0 :::8042                 :::*                    LISTEN      -
tcp6       0      0 :::3306                 :::*                    LISTEN      -
tcp6       0      0 :::2222                 :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
```

此时可以通过jdbc连接hive或mysql使用测试环境中的数据。

## 关闭测试环境

进入metric-sys-test目录，执行：

```
$ docker-compose stop
```

## 重新启动测试环境

进入metric-sys-test目录，执行：

\$ docker-compose start

SHELL

等待一会儿，直到观察到端口10000被占用：

\$ netstat -tlnp

SHELL

(Not all processes could be identified, non-owned process info  
will not be shown, you would have to be root to see it all.)

Active Internet connections (only servers)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State	PID/Program name
tcp	0	0	127.0.0.1:25	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:50010	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:50075	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:50020	0.0.0.0:*	LISTEN	-
tcp	0	0	127.0.0.1:9000	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:50090	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:2222	0.0.0.0:*	LISTEN	-
tcp	0	0	127.0.0.1:32975	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:10000	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:10002	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:50070	0.0.0.0:*	LISTEN	-
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN	-
tcp6	0	0	:::8088	:::*	LISTEN	-
tcp6	0	0	:::1:25	:::*	LISTEN	-
tcp6	0	0	:::13562	:::*	LISTEN	-
tcp6	0	0	:::8030	:::*	LISTEN	-
tcp6	0	0	:::8031	:::*	LISTEN	-
tcp6	0	0	:::8032	:::*	LISTEN	-
tcp6	0	0	:::8033	:::*	LISTEN	-
tcp6	0	0	:::8040	:::*	LISTEN	-
tcp6	0	0	:::40296	:::*	LISTEN	-
tcp6	0	0	:::8042	:::*	LISTEN	-
tcp6	0	0	:::3306	:::*	LISTEN	-
tcp6	0	0	:::2222	:::*	LISTEN	-
tcp6	0	0	:::22	:::*	LISTEN	-

此时可以通过jdbc连接hive或mysql使用测试环境中的数据。

Version 1.0

Last updated 2020-07-23 10:47:26 +0800