swarm使用TIG监控

--

grafana docker swarm 监控官方文档： *https://grafana.com/grafana/dashboards/3674&https://github.com/mlabouardy/swarm-tig*
用已有的grafana+ yum安装influxdb + docker全局部署telegraf
grafana安装省略

1. influxdb 安装

```
wget https://dl.influxdata.com/influxdb/releases/influxdb-1.1.0.x86_64.rpm
rpm -ivh influxdb-1.1.0.x86_64.rpm
systemctl enable influxdb
systemctl start influxdb
```

创建数据库：

```
[root@app058 stack]# influx
Visit https://enterprise.influxdata.com to register for updates, InfluxDB server management, and monitoring.
Connected to http://localhost:8086 version 1.1.0
InfluxDB shell version: 1.1.0
> create database docker_metrics;
> create database vm_metrics;
> quit
```

2. 编写telegraf.conf文件

```
[global_tags]
environment="swarm"

# Read metrics about CPU usage
[[inputs.cpu]]
  percpu = false
  totalcpu = true
  fieldpass = [ "usage*" ]
  name_suffix = "_vm"

# Read metrics about disk usagee
[[inputs.disk]]
```

```
  fielddrop = [ "inodes*" ]
  mount_points=["/"]
  name_suffix = "_vm"

# Read metrics about network usage
[[inputs.net]]
  interfaces = [ "eth0" ]
  fielddrop = [ "icmp*", "ip*", "tcp*", "udp*" ]
  name_suffix = "_vm"

# Read metrics about memory usage
[[inputs.mem]]
  name_suffix = "_vm"

# Read metrics about swap memory usage
[[inputs.swap]]
  name_suffix = "_vm"

# Read metrics about system load & uptime
[[inputs.system]]
  name_suffix = "_vm"

# Read metrics from docker socket api
[[inputs.docker]]
  endpoint = "unix:///var/run/docker.sock"
  container_names = []
  name_suffix = "_docker"

[[outputs.influxdb]]
  database = "vm_metrics"
  # 修改成 Influxdb的地址
  urls = ["http://172.16.1.188:8086"]
  namepass = ["*_vm"]

[[outputs.influxdb]]
  database = "docker_metrics"
  urls = ["http://172.16.1.188:8086"]
  namepass = ["*_docker"]
```

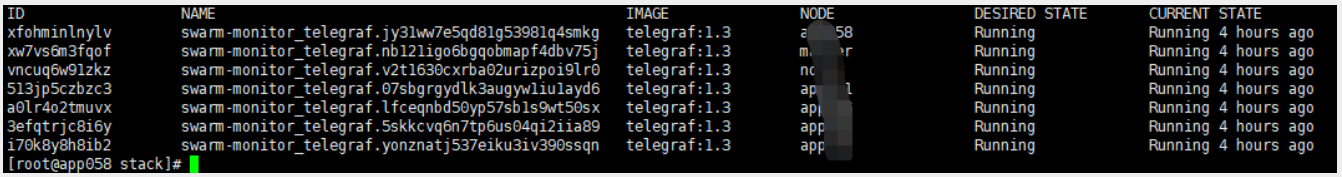
3. 编写swarm-monitor.yml，与telegraf.conf文件目录同级

```yaml
version: "3.4"

services:
  telegraf:
    image: telegraf:1.3
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock
    configs:
      - source: telegraf-config
        target: /etc/telegraf/telegraf.conf
    deploy:
      restart_policy:
        condition: on-failure
      mode: global

configs:
  telegraf-config:
    file: $PWD/telegraf.conf
```

4. swam 全局部署

   在swarm manager节点上执行： `docker stack deploy -c swarm-monitor swarm-monitor`

   查看部署信息：`docker stack ps swarm-monitor`

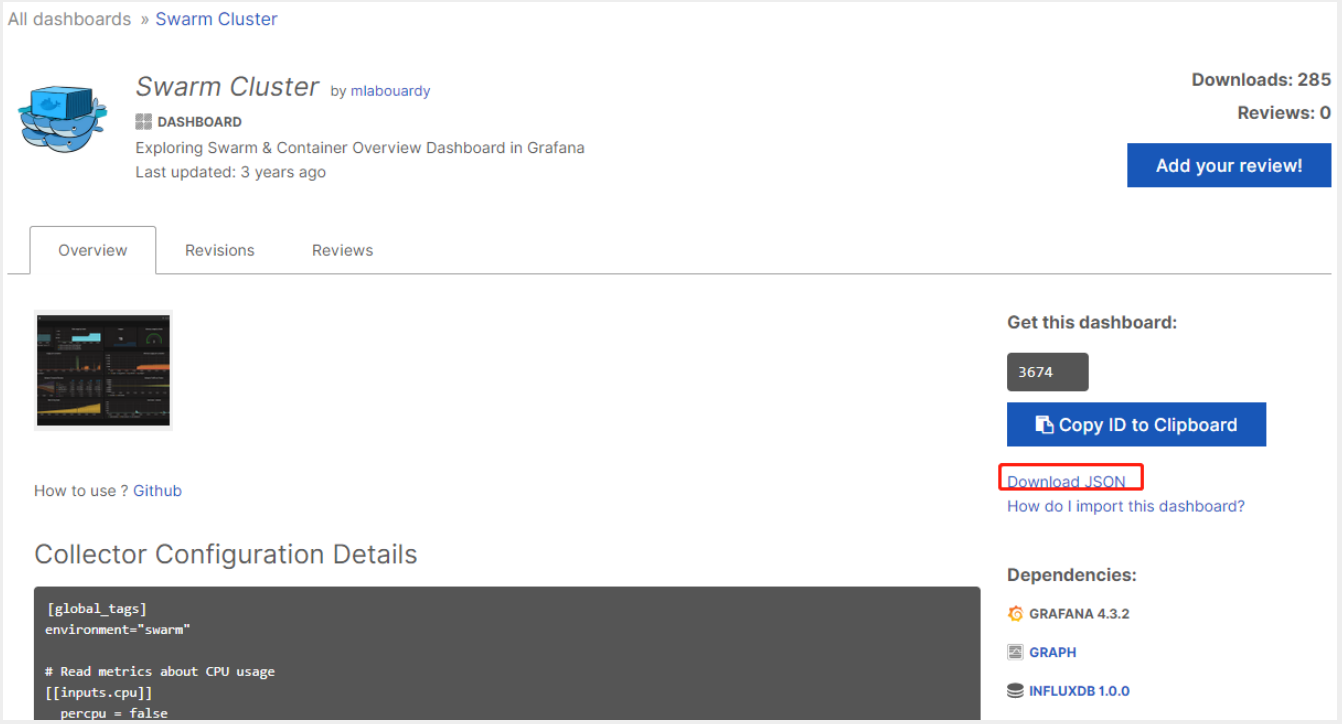


```
ID                  NAME                                                IMAGE               NODE                DESIRED STATE       CURRENT STATE
xfohminlnylv        swarm-monitor_telegraf.jy31ww7e5qd81g53981q4smkg    telegraf:1.3        a[illegible]58      Running             Running 4 hours ago
xw7vs6m3fqof        swarm-monitor_telegraf.nb12ligo6bgqobmapf4dbv75j    telegraf:1.3        m[illegible]r       Running             Running 4 hours ago
vncuq6w91zkz        swarm-monitor_telegraf.v2tl630cxrba02urizpoi9lr0    telegraf:1.3        no[illegible]       Running             Running 4 hours ago
513jp5czbzc3        swarm-monitor_telegraf.07sbgrgydlk3augyw1iu1ayd6    telegraf:1.3        ap[illegible]l      Running             Running 4 hours ago
a0lr4o2tmuvx        swarm-monitor_telegraf.lfceqnbd50yp57sb1s9wt50sx    telegraf:1.3        ap[illegible]       Running             Running 4 hours ago
3efqtrjc8i6y        swarm-monitor_telegraf.5skkcvq6n7tp6us04qi2iia89    telegraf:1.3        app[illegible]      Running             Running 4 hours ago
i70k8y8h8ib2        swarm-monitor_telegraf.yonznatj537eiku3iv390ssqn    telegraf:1.3        app[illegible]      Running             Running 4 hours ago
[root@app058 stack]#
```

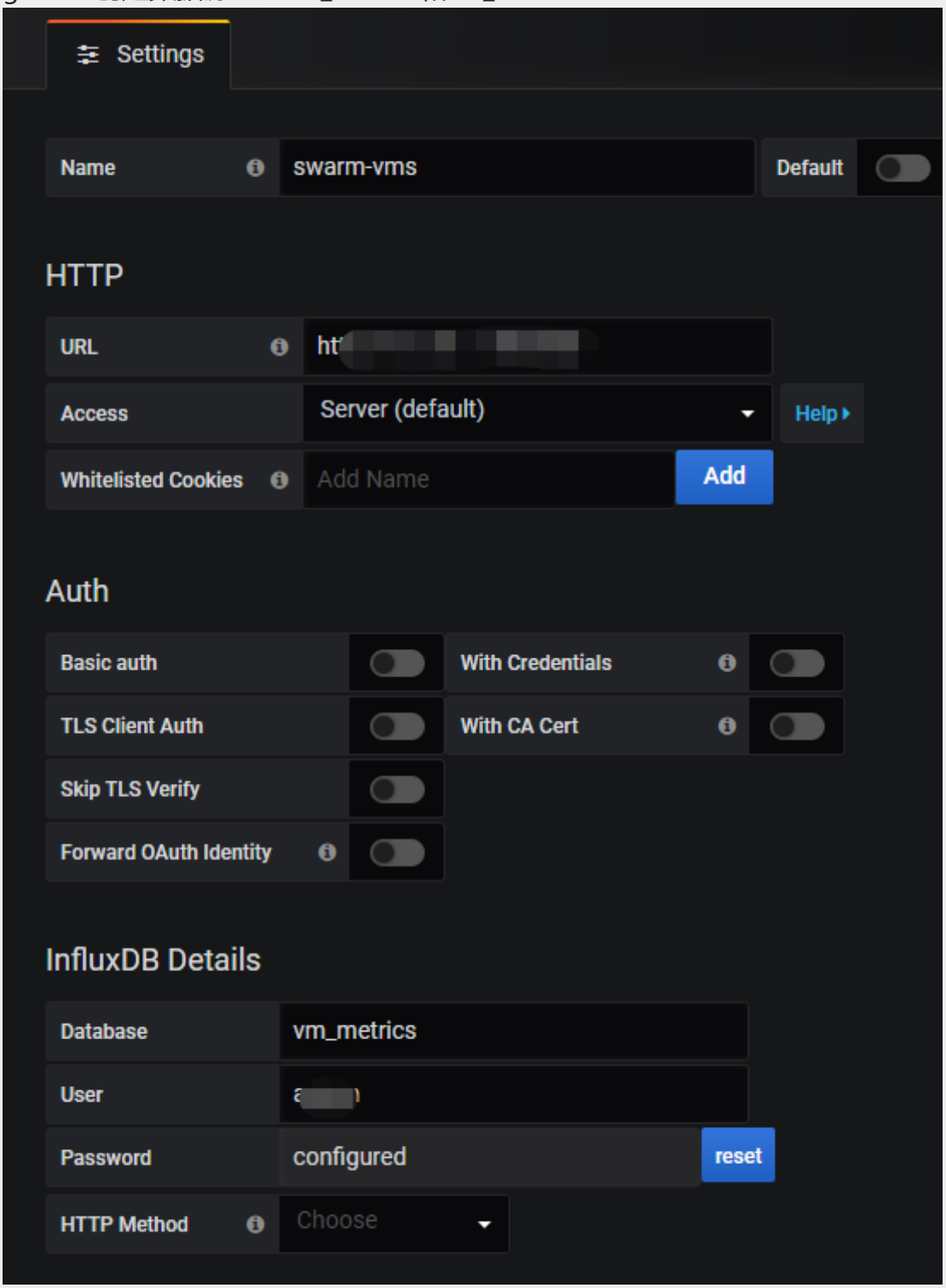
   如果出现异常可以进入容器查看具体内容：`docker logs -f 容器ID`

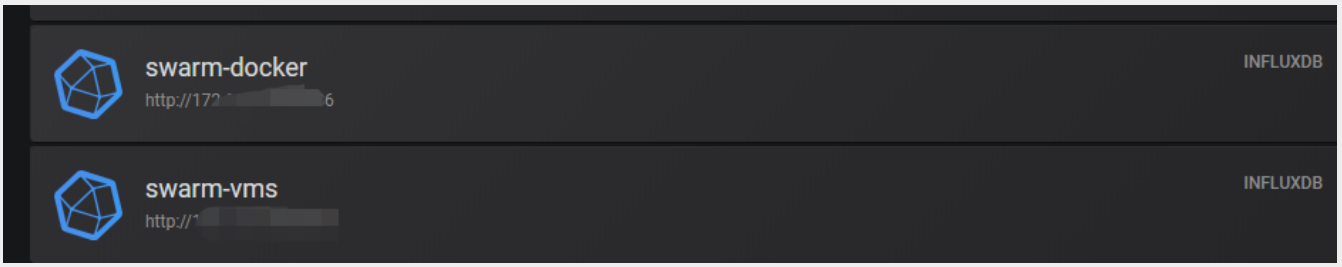
```
[root@app058 stack]# docker logs -f 1db31b231a9b
2020/06/01 10:58:11 I! Using config file: /etc/telegraf/telegraf.conf
2020-06-01T10:58:11Z I! Starting Telegraf (version 1.3.5)
2020-06-01T10:58:11Z I! Loaded outputs: influxdb influxdb
2020-06-01T10:58:11Z I! Loaded inputs: inputs.net inputs.mem inputs.swap inputs.system inputs.docker inputs.cpu inputs.disk
2020-06-01T10:58:11Z I! Tags enabled: environment=swarm host=1db31b231a9b
2020-06-01T10:58:11Z I! Agent Config: Interval:10s, Quiet:false, Hostname:"1db31b231a9b", Flush Interval:10s
```

5. dashbaord文件下载地址

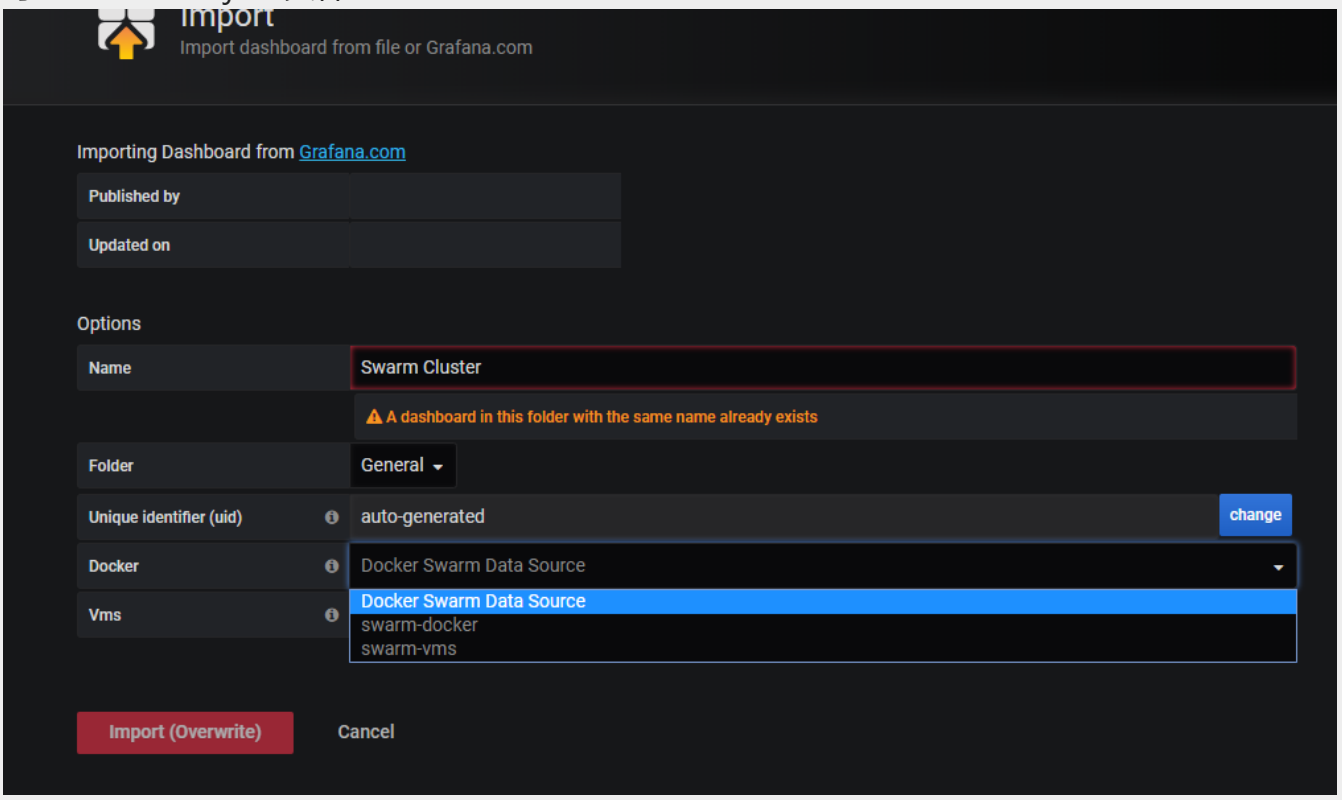


6. grafana创建数据源 docker_metrics和vm_metrics





7. 导入dashboard.json文件



8. 监控情况：