Openstack监控研究报告

苏逸民

一、关键问题

1. 监控云硬件和服务

硬件信息包括磁盘空间使用率、磁盘IO、CPU使用率、内存使用率等。服务包括mysql、rabbitmq、libvirt以及Openstack的服务进程。

1. 租户的资源

包括监控租户的虚拟机数目和状态、浮动IP数目、虚拟磁盘数目和容量等。

1. 监控虚拟机的性能

包括虚拟机的磁盘空间使用率、磁盘IO、CPU使用率、内存使用率等。

二、拟采用的技术路线

1. 使用Nagios收集数据。

Nagios已有的插件能够收集物理机的性能信息，而Openstack的租户的资源和虚拟机性能信息则需要自己写插件调用Openstack的服务来获取。Openstck的ceilometer服务提供了虚拟机的性能信息，安装配置后，可以方便地通过Openstack的SDK获取这些性能信息。

1. 使用graphite做数据持久化

Nagios本身不提供数据持久化的功能，因此也不能看到历史数据，更加无法看到图表。Graphite使用类似RRDTool的方式存储这些数据，但不负责如何收集数据。因此需要Nagios收集数据并提供给Graphite存储，Graphios则负责把Nagios收集到的性能数据存储到Graphite中。

1. 渲染图表

虽然Graphite也能渲染图表，但还不够好。Grafana能出色地完成渲染图表的功能，并且允许用户轻松地操作界面，控制仪表盘要展现的图表。而配置也相当简单，只需要把数据源指定为Graphite即可。

1. 使用docker打包

隐藏所有配置的细节，简化部署，让管理员能够使用一条命令安装整个技术栈。

三、技术准备

1. 学习Nagios，使用Nagios的插件监控不同主机和服务，并学习如何编写插件。

2. 了解Openstack的SDK的使用方法。

3. 学习Graphite的配置方法。

四、关键技术验证

以下都在ubuntu server 14.04 测试

1. 安装配置Nagios
   1. 安装LAMP和Nagios

sudo apt-get install -y apache2

sudo apt-get install -y mysql-server mysql-client

sudo apt-get install -y php5 php5-mysql libapache2-mod-php5

sudo apt-get install -y nagios3 nagios-nrpe-plugin

* 1. 配置Nagios

sudo usermod -a -G nagios www-data

sudo chmod -R +x /var/lib/nagios3/

修改 /etc/nagios3/nagios.cfg

check\_external\_commands=1

重启nagios3服务后，可以通过浏览器访问host\_url/nagios3

* 1. 配置要监控的机器

sudo apt-get install nagios-nrpe-server nagios-plugins

编辑 /etc/nagios/nrpe.cfg

allowed\_hosts=127.0.0.1,10.10.1.81

其中10.10.1.81为运行nagios服务的机器的IP

重启nagios-nrpe-server服务

* 1. 修改nagios配置

在运行nagios的机器上创建文件/etc/nagios3/conf.d/servers.cfg，内容为

define host{

use generic-host

host\_name target

alias target

address 10.10.1.82

max\_check\_attempts 5

check\_period 4x7

notification\_interval 30

notification\_period 24x7

}

保存并重启nagios，在web界面可以看到新添加的主机

* 1. 添加监控项目

定义监控的命令，创建/etc/nagios3/conf.d/commands.cfg

define command {

command\_name mycheck\_load

command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c check\_load

}

这条命令会使用check\_nrpe调用远程主机上的check\_load命令

在/etc/nagios3/conf.d/servers.cfg添加如下内容

define service {

use generic-service

host\_name target

service\_description cpu load

check\_command mycheck\_load

}

保存并重启nagios，在web界面可以看到新添加的服务

1. 安装配置Graphite
   1. 安装Graphite

sudo apt-get install apache2 libapache2-mod-wsgi python-django python-twisted python-cairo python-pip python-django-tagging

sudo pip install pytz

sudo pip install whisper  
sudo pip install carbon  
sudo pip install graphite-web

* 1. 配置Graphite

在/opt/graphite/conf目录下执行下面命令

sudo cp carbon.conf.example carbon.conf  
sudo cp storage-schemas.conf.example storage-schemas.conf  
sudo cp graphite.wsgi.example graphite.wsgi

配置Apache

sudo cp /opt/graphite/example/example-graphite-vhost.conf  \

/etc/apache2/sites-available/graphite-vhost.conf

修改文件内容为

Listen 8008

<IfModule ....>

...

WSGISocketPrefix /var/run/apache2/wsgi

<VirtualHost \*:8008>...

在/etc/apache2/sites-enabled创建软链接

cd /etc/apache2/sites-enabled

sudo ln -s ../sites-available/graphite-vhost.conf   graphite-vhost.conf

注释掉/etc/apache2/apache2.conf第156行

<Directory>

Opitons ...

AllowOverride None

# Require all denied # comment this line

</Directory>

初始化数据库

cd /opt/graphite/webapp/graphite/

sudo python manage.py syncdb  
sudo chown -R www-data:www-data /opt/graphite/storage/   
sudo cp local\_settings.py.example local\_settings.py

sudo /etc/init.d/apache2 restart

启动carbon

/opt/graphite/bin/carbon-cache.py start

通过浏览器访问8008端口可以看到Graphite的界面

* 1. 使用Graphios把Nagios的性能数据存储到Graphite

安装Graphios

git clone https://github.com/shawn-sterling/graphios.git

cd graphios

python setup.py install

mkdir -p /var/spool/nagios/graphios

chown nagios:nagios -R /var/spool/nagios

配置

在/etc/nagios3/conf.d/commands.cfg加入以下内容

define command {

command\_name graphite\_perf\_host

command\_line /bin/mv \

/var/spool/nagios/graphios/host-perfdata \

/var/spool/nagios/graphios/host-perfdata.$TIMET$

}

define command {

command\_name graphite\_perf\_service

command\_line /bin/mv \

/var/spool/nagios/graphios/service-perfdata \

/var/spool/nagios/graphios/service-perfdata.$TIMET$

}

将之前配置Nagios时的/etc/nagios3/conf.d/servers.cfg内容改为

define service {

use generic-service

host\_name target

service\_description cpu load

check\_command mycheck\_load

\_graphiteprefix Moniter.cpuload

}

修改 /etc/graphios/graphios.cfg的配置

enable\_carbon = True

修改 /usr/local/bin/graphios.py

config\_file = '/etc/graphios/graphios.cfg'

创建文件/usr/local/nagios/var/graphios.log

重启服务

service graphios start

service nagios restart

稍等几分钟就能够在Graphite的web界面看到图表。

1. 安装配置Grafana

echo ‘deb https://packagecloud.io/grafana/stable/debian/ jessie main’ >> /etc/apt/sources.list

curl https://packagecloud.io/gpg.key | sudo apt-key add -

sudo apt-get update

sudo apt-get install grafana

重启grafana-server服务，通过端口3000可以访问grafana

使用账号密码admin：admin登录Grafana的web界面，并添加数据源为Graphite。可以随意地添加图表到仪表盘，并为图表设置数据源、大小、样式。之前配置的CPU load也可以通过仪表盘展示了。

1. 自定义Nagios插件监控openstack虚拟机
   1. 编写Nagios插件，插件调用ceilometer的客户端获取监控采样。代码如下：

import ceilometerclient.client

auth\_args = {

'auth\_url': 'http://10.10.1.11:35357/v3',

'project\_name': 'admin',

'user\_domain\_name': 'default',

'project\_domain\_name': 'default',

'username': 'admin',

'password': 'admin123',

}

cclient = ceilometerclient.client.get\_client(2,

os\_username=auth\_args['username'],

os\_password=auth\_args['password'],

os\_tenant\_name=auth\_args['project\_name'],

os\_user\_domain\_name=auth\_args['user\_domain\_name'],

os\_project\_domain\_name=auth\_args['project\_domain\_name'],

os\_auth\_url=auth\_args['auth\_url'])

query = [

dict(field='resource\_id', op='eq',

value='f0bb0224-dab6-4531-bf63-94fe2d5b7686'),

dict(field='meter',op='eq',value='cpu\_util')

]

for sample in cclient.new\_samples.list(q=query, limit=1):

print 'util %s|cpu\_util=%s;;;;' % (sample.volume, sample.volume)

把脚本移动到 /usr/lib/nagios/plugins/check\_vm.py

* 1. 修改Nagios配置

添加命令

define command {

command\_name check\_vm

command\_line /usr/bin/python /usr/lib/nagios/plugins/check\_vm.py

}

添加服务，其中mitaka为openstack的控制节点

define service {

Use generic-service

host\_name mitaka

service\_description check-vm

check\_command check\_vm

normal\_check\_interval 1

\_graphiteprefix Moniter.vm

}

重启nagios3后可以在Grafana的界面上配置对应的图表。

到这里所有关键技术到验证完毕。