

zabbix监控（四）监控全网服务器

原创

GeorgeKai

2018-03-20 21:47:51

评论(0)

736人阅读

监控全网服务器

5.1 需求说明

实际需求：

公司已经有了100台服务器，现在需要使用zabbix全部监控起来。

5.2 规划方案

常规监控：cpu，内存，磁盘，网卡 问题：怎样快速添加100台机器

方法1：使用克隆的方式

方法2：自动注册和自动发现

方法3：调用zabbix api接口 curl 、python

开发自己的运维平台兼容zabbix的通道

服务监控，url监控等特殊监控：自定义监控

5.2.1 api接口使用（curl）

```
curl -i -X POST -H 'Content-Type:application/json' -d '{"jsonrpc":
"2.0","method":"user.login","params":{"user":"Admin","password":"zabbix"},"auth":
null,"id":0}' "http://10.0.0.61/zabbix/api_jsonrpc.php"
curl -i -X POST -H 'Content-Type:application/json' -d'
{
"jsonrpc": "2.0",
"method": "host.get",
"params": {
"output": [
"hostid",
"host"
],
"selectInterfaces": [
"interfaceid",
"ip"
]
},
"id": 2,
"auth": "6a450a8fc3dce71fd310cfe338746578"
}' "http://10.0.0.61/zabbix/api_jsonrpc.php"
```

5.3 具体实施规划

5.3.1 硬件、系统、网络监控

所有集群节点（所有虚拟机）都监控上

交换机，路由器监控（简单方法：换成端口对应服务器网卡流量监控；标准方法：监控交换机的网卡）

snmp监控

5.3.2 应用服务监控

1. 监控备份服务器，简单方法是监控rsync端口，如果有其他更佳方案可以说明；

方法1：监控873端口 `net.tcp.port[,873]`

方法2：模拟推送拉取文件

方法3：监控进程数

2. 监控NFS服务器，使用监控NFS进程来判断NFS服务器正常，如果有其他更佳方案可以说明；

方法1：端口（通过111的rpc端口获取nfs端口） `net.tcp.port[,111]`

方法2： `showmount -e ip|wc -l`

3. 监控MySQL服务器，简单方法监控mysql的3306端口，或者使用zabbix提供的Mysql模板，如果有其他更佳方案可以说明；

方法1：端口（通过3306的mysql端口） `net.tcp.port[,3306]`

方法2：mysql远程登录

方法3：使用zabbix agent自带的模板及key

4. 监控2台web服务器，简单方法监控80端口，如果有其他更佳方案可以说明；

方法1：端口（通过80的web端口） `net.tcp.port[,80]`

方法2：看网页状态码、返回内容==zabbix 自带WEB检测

5. 监控URL地址来更精确的监控我们的网站运行正常；

使用zabbix自带的监控Web监测 进行监控

6. 监控反向代理服务器，PPTP服务器等你在期中架构部署的服务。

nginx, pptp

ntp 端口udp 123

7. 监控Nginx的7种连接状态。

自定义监控

5.3.3 监控服务通用方法

1. 监控端口 `netstat ss lsof ==》 wc -l`

2. 监控进程 `ps -ef|grep 进程|wc -l` 试运行一下

3. 模拟客户端的使用方式监控服务端

web ==》 curl

mysql ==》 select insert

memcache ==》 set再get

5.4 实施全网监控

安装客户端脚本，for centos6

```
#!/bin/bash
```

#设置解析

```
# echo '10.0.0.1 mirrors.aliyuncs.com mirrors.aliyun.com repo.zabbix.com' >> /etc/hosts
```

#安装zabbix源、aliyu nYUM源

```
curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo
```

```
curl -o /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-6.repo
```

```
rpm -ivh http://repo.zabbix.com/zabbix/3.0/rhel/6/x86_64/zabbix-release-3.0-1.el6.noarch.rpm
```

```
yum clean all
```

```
yum clean all
```

#安装zabbix客户端

```
yum install zabbix-agent -y
```

```
sed -i.ori 's#Server=127.0.0.1#Server=172.16.1.61#' /etc/zabbix/zabbix_agentd.conf
```

```
/etc/init.d/zabbix-agent start
```

#写入开机自启动

```
chmod +x /etc/rc.d/rc.local
```

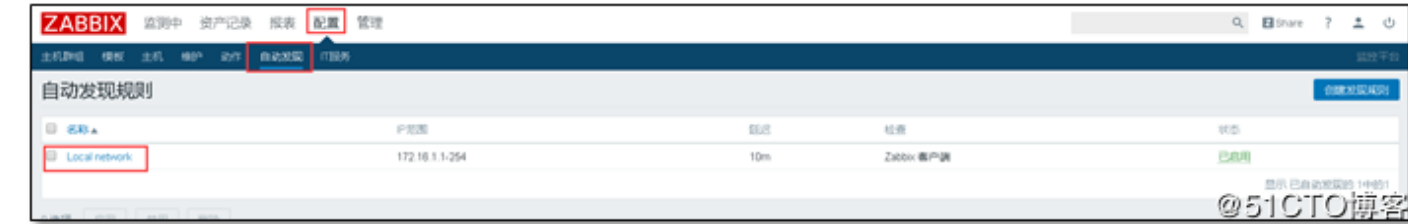
```
cat >>/etc/rc.d/rc.local<<EOF
```

```
/etc/init.d/zabbix-agent start
```

```
EOF
```

5.4.1 使用自动发现规则

添加自动发现规则



创建发现动作



查看自动发现的机器。

| 已发现的设备 | 已监测的主机 | 在线时间/离线时间 | Zabbix |
|-----------------------|--------------------------------------|-----------|--------|
| Local network (11个设备) | | | |
| 172.16.1.5 | 负载均衡(10.0.0.5/172.16.1.5) | 01:13:16 | Online |
| 172.16.1.6 | 负载均衡(10.0.0.6/172.16.1.6) | 01:13:16 | Online |
| 172.16.1.7 | web02服务器(10.0.0.7/172.16.1.7) | 00:57:53 | Online |
| 172.16.1.8 | web01服务器(10.0.0.8/172.16.1.8) | 01:13:11 | Online |
| 172.16.1.9 | web03服务器(10.0.0.9/172.16.1.9) | 00:57:52 | Online |
| 172.16.1.21 | cache01服务器(10.0.0.21/172.16.1.21) | 00:57:18 | Online |
| 172.16.1.31 | NFS服务器(10.0.0.31/172.16.1.31) | 01:11:37 | Online |
| 172.16.1.41 | 备份服务器(10.0.0.41/172.16.1.41) | 00:56:12 | Online |
| 172.16.1.51 | mysql数据库(10.0.0.51/172.16.1.51) | 01:10:32 | Online |
| 172.16.1.61 (m01) | Zabbix server(10.0.0.61/172.16.1.61) | 01:25:14 | Online |
| 172.16.1.81 | nfs高可用(10.0.0.81/172.16.1.81) | 01:08:47 | Online |

5.4.2 监控备份服务器

利用系统自带键值进行监控net.tcp.listen[port] 创建新的模板



在服务端进行测试

```
[root@m01 ~]# zabbix_get -s 172.16.1.41 -p 10050 -k "net.tcp.listen[873]"
1
# 1为端口在监听 0为端口未监听
#通过进程监控
proc.num[<name>,<user>,<state>,<cmdline>] 进程数。返回整数
[root@m01 ~]# zabbix_get -s 172.16.1.21 -p 10050 -k "proc.num[zabbix_agentd,,]"
6
[root@m01 ~]# zabbix_get -s 172.16.1.21 -p 10050 -k "proc.num[,root,,]"
330
[root@m01 ~]# zabbix_get -s 172.16.1.21 -p 10050 -k "proc.num[,root,sleep,]"
330
[root@m01 ~]# zabbix_get -s 172.16.1.21 -p 10050 -k "proc.num[,sleep,]"
339
```

将模板添加到主机



5.4.3 监控NFS服务器

创建nfs监控模板

方法一：监控rpcbind的111端口

方法二：nfs进程

使用 `proc.num[<name>.<user>.<state>.<cmdline>]` 键值，检测nfs进程的数量

在服务端进行测试

```
[root@m01 ~]# zabbix_get -s 172.16.1.31 -p 10050 -k "proc.num[,,,rpc]"
5
[root@m01 ~]# zabbix_get -s 172.16.1.31 -p 10050 -k "proc.num[nfsd,,,"]
8
```

将模板绑定到主机

5.4.4 监控MySQL服务器

使用系统自带模板 `net.tcp.port[<ip>,<port>]` 利用自带的监控端口键值进行监控

添加新的mysql监控项端口

```
[root@m01 ~]# zabbix_get -s 172.16.1.51 -p 10050 -k "net.tcp.port[,3306]"
1
#检查是否能建立 TCP 连接到指定端口。返回 0 - 不能连接; 1 - 可以连接
将模板关联到主机
```

方法二：使用zabbix提供的mysql模板

`/etc/zabbix-agent.d/`下自带mysqlkey

要使用自带的mysqlkey值加上mysql的账户密码，否则不能获取到数据。

1.修改zabbix-agent.d/userparameter_mysql.conf

```
# Flexible parameter to grab global variables. On the frontend side, use keys like mysql.status[Com_insert].
# Key syntax is mysql.status[variable].
UserParameter=mysql.status[*],echo "show global status where Variable_name='$1';" | HOME=/var/lib/zabbix mysql -u
root -N | awk '{print $$2}'
```

2.重启服务

```
systemctl restart zabbix-agent.service
```

3.测试key值

```
[root@m01 ~]# zabbix_get -s 172.16.1.51 -p 10050 -k "mysql.ping"
1
[root@m01 ~]# zabbix_get -s 172.16.1.51 -p 10050 -k "mysql.status[Com_begin]"
3758
[root@m01 ~]# zabbix_get -s 172.16.1.51 -p 10050 -k "mysql.status[Com_delete]"
3
```

4.网页m01关联模板Template App MySQL



@51CTO博客

5.4.5 监控web服务器

方法1：创建监控模板 监控 nginx服务与 80 端口

proc.num[<name>,<user>,<state>,<cmdline>] 进程数。返回整数
net.tcp.port[<ip>,<port>] 检查是否能建立 TCP 连接到指定端口。返回 0 - 不能连接；1 - 可以连接



@51CTO博客

```
[root@m01 ~]# zabbix_get -s 172.16.1.8 -p 10050 -k "proc.num[,,,nginx]"
2
[root@m01 ~]# zabbix_get -s 172.16.1.8 -p 10050 -k "net.tcp.port[,80]"
1
```

将模板关联到主机

| | | | | | | | | | |
|-------------------------------------|-------------------------------|--------|--------|--------|------|--------|-------|-------------------|---|
| <input checked="" type="checkbox"/> | web03服务器(10.0.0.9/172.16.1.9) | 应用集 11 | 监控项 48 | 触发器 21 | 图形 9 | 自动发现 2 | Web监测 | 172.16.1.9: 10050 | Template App Nginx, Template OS Linux (Template App Zabbix Agent) |
| <input checked="" type="checkbox"/> | web01服务器(10.0.0.8/172.16.1.8) | 应用集 11 | 监控项 48 | 触发器 21 | 图形 9 | 自动发现 2 | Web监测 | 172.16.1.8: 10050 | Template App Nginx, Template OS Linux (Template App Zabbix Agent) |
| <input checked="" type="checkbox"/> | web02服务器(10.0.0.7/172.16.1.7) | 应用集 11 | 监控项 48 | 触发器 21 | 图形 9 | 自动发现 2 | Web监测 | 172.16.1.7: 10050 | Template App Nginx, Template OS Linux (Template App Zabbix Agent) |

@51CTO博客

方法二：检测URL地址

ZABBIX

监测中 资产记录 报表 配置 管理

主机群组 模板 主机 维护 动作 自动发现 IT服务

老男孩教育监控平台

主机

过滤器

名字是 DNS名 IP地址 端口

过滤 重设

| 名称 | 应用集 | 监控项 | 触发器 | 图形 | 自动发现 | Web监测 | 端口 | 模板 | 状态 | 可用性 | agent | 加密 | 信息 |
|---|--------|--------|--------|-------|--------|-------|--------------------|---|-----|-------------------|-------|----|----|
| Memcache (10.0.0.21/172.16.1.21) | 应用集 11 | 监控项 49 | 触发器 20 | 图形 10 | 自动发现 2 | Web监测 | 172.16.1.21: 10050 | Template Login, Template OS Linux (Template App Zabbix Agent) | 已启用 | ZBX SNMP JMX IPMI | 无 | | |
| Zabbix server (10.0.0.61/172.16.1.61) | 应用集 12 | 监控项 91 | 触发器 47 | 图形 16 | 自动发现 2 | Web监测 | 172.16.1.61: 10050 | Template App MySQL, Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent) | 已启用 | ZBX SNMP JMX IPMI | 无 | | |

显示 已自动发现的 2中的2

0 选择 启用 禁用 导出 批量更新 删除

Memcache (10.0.0.21/172.16.1.21)

应用集 11 监控项 49 触发器 20 图形 10 自动发现 2

Web监测 10050

Template Login, Template OS Linux (Template App Zabbix Agent)

已启用 ZBX SNMP JMX IPMI 无

Zabbix server (10.0.0.61/172.16.1.61)

应用集 12 监控项 91 触发器 47 图形 16 自动发现 2

Web监测 172.16.1.61: 10050

Template App MySQL, Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)

已启用 ZBX SNMP JMX IPMI 无

显示 已自动发现的 2中的2

0 选择 启用 禁用 导出 批量更新 删除

Web监测

过滤器

状态 所有 已启用 停用的

过滤 重设

| 名称 | 步骤数量 | 更新间隔 | 尝试次数 | 认证 | HTTP代理 | 应用集 | 状态 | 信息 |
|-------|------|------|------|----|--------|-----|----|----|
| 未发现数据 | | | | | | | | |

显示 已自动发现的 0中的0

0 选择 启用 禁用 清除历史 删除

场景 步骤 认证

名称 监控首页

应用集

新的应用集 Web监测

数据更新间隔(秒) 10

尝试次数 1

客户端 Zabbix

HTTP代理 http://[user[:password]@]proxy.example.com[:port]

变量

头

已启用

Web监测

所有主机 / Zabbix server (10.0.0.61/172.16.1.61) 已启用 ZBX SNMP JMX IPMI 应用集 12 监控项 91 触发器 47 图形 16 自动发现规则 2 Web 场景

场景 步骤 认证

| 步骤 | 名称 | 超时 | URL | 要求的 | 状态码 | 动作 |
|----|----|----|-----|-----|-----|----|
| | 添加 | | | | | |

添加 取消

@51CTO博客

名称 监控第一个页面

URL http://10.0.0.61/

Post

变量

头

跟随跳转 ☒

仅获取头信息 ☐

超时 15

要求的字符串 Testing

要求的状态码 200

添加 取消

@51CTO博客

Web监测

所有主机 / Zabbix server (10.0.0.61/172.16.1.61) 已启用 ZBX SNMP JMX IPMI 应用集 12 监控项 91 触发器 47 图形 16 自动发现规则 2 Web 场

场景 步骤 认证

| 步骤 | 名称 | 超时 | URL | 要求的 | 状态码 | 动作 |
|----|---------|------|-------------------|---------|-----|----|
| 1: | 监控第一个页面 | 15 秒 | http://10.0.0.61/ | Testing | 200 | 移除 |

添加 取消

@51CTO博客

Web监测

所有主机 / Zabbix server (10.0.0.61/172.16.1.61) 已启用 ZBX SNMP JMX IPMI 应用集 12 监控项 91 触发器 47 图形 16 自动发现规则 2 Web 场景

场景 步骤 认证

步骤

| 名称 | 超时 | URL | 要求的 | 状态码 | 动作 |
|------------|------|-------------------|---------|-----|--------------------|
| 1: 监控第一个页面 | 15 秒 | http://10.0.0.61/ | Testing | 200 | 移除 |

添加

取消

ZABBIX

监测中 资产记录 报表 配置 管理

仪表盘 概述 Web监测 最新数据 触发器 事件 图形 聚合图形 拓扑图 自动发现 IT服务

Web监测

主机 名称 步骤数量 最近检查记录 状态

| | | | | |
|---|------|---|---------------------|----|
| Zabbix server (10.0.0.61/172.16.1.61) | 监控首页 | 1 | 2018-03-16 09:50:29 | 正常 |
|---|------|---|---------------------|----|

显示 已自动发现的 1中的1

测试：

ZABBIX

监测中 资产记录 报表 配置 管理

仪表盘 概述 Web监测 最新数据 触发器 事件 图形 聚合图形 拓扑图 自动发现 IT服务

Web监测

主机 名称 步骤数量 最近检查记录 状态

| | | | | |
|---|------|---|---------------------|----|
| Zabbix server (10.0.0.61/172.16.1.61) | 监控首页 | 1 | 2018-03-16 09:50:09 | 正常 |
|---|------|---|---------------------|----|

显示 已自动发现的 1中的1

ZABBIX

监测中 资产记录 报表 配置 管理

仪表盘 概述 Web监测 最新数据 触发器 事件 图形 聚合图形 拓扑图 自动发现 IT服务

Web监测

主机 名称 步骤数量 最近检查记录 状态

| | | | | |
|---|------|---|---------------------|----|
| Zabbix server (10.0.0.61/172.16.1.61) | 监控首页 | 1 | 2018-03-16 09:50:29 | 正常 |
|---|------|---|---------------------|----|

显示 已自动发现的 1中的1

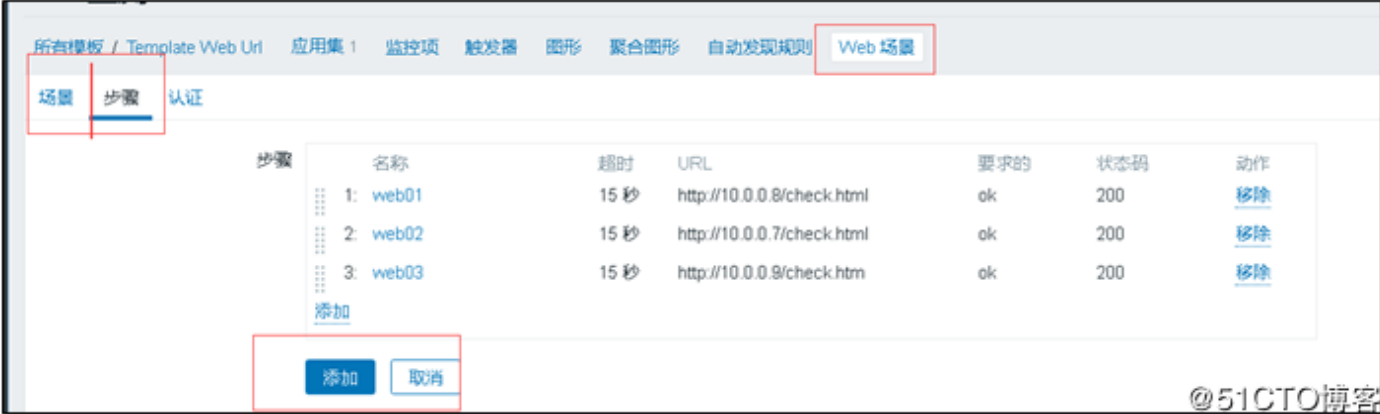
5.4.6 监控URL地址

创建监测页面
echo ok >> /application/nginx/html/www/check.html

测试监控页面
[root@web03 ~]# for ip in 7 8 9 ;do curl 10.0.0.\$ip/check.html ;done
ok
ok
创建web监测模板
创建应用集



创建Web场景



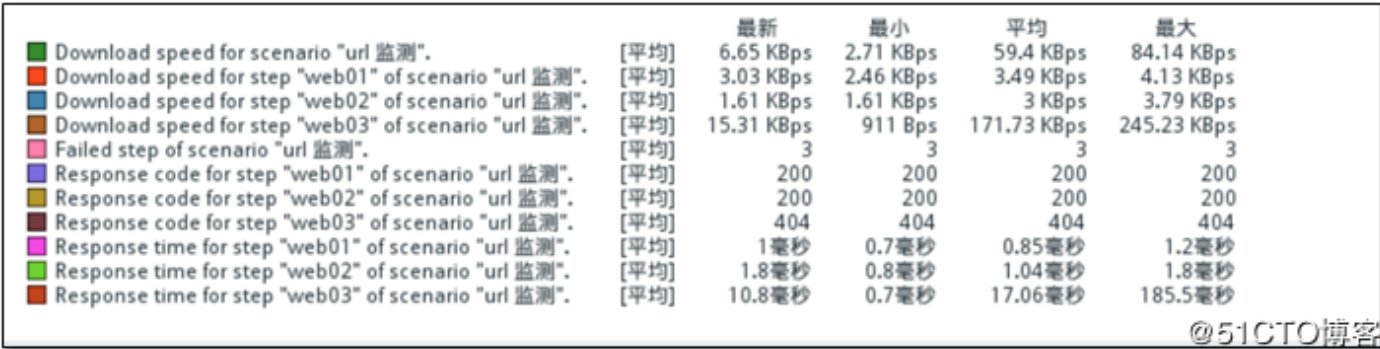
创建图形



将模板关联到主机



监测结果



5.4.7 监控反向代理服务器

创建自定义key

```
[root@lb01 ~]# cat /etc/zabbix/zabbix_agentd.d/userparameter_nk.conf
```

```
UserParameter=keep-ip,ip a |grep 10.0.0.3|wc -l
```

在服务端测试

```
[root@m01 ~]# zabbix_get -s 172.16.1.5 -p 10050 -k "keep-ip"
```

```
1
```

```
[root@m01 ~]# zabbix_get -s 172.16.1.6 -p 10050 -k "keep-ip"
```

```
0
```

在web界面添加模板



将模板关联到主机



5.4.8 监控Nginx的7种连接状态

nginx服务器显示status

```
.....
```

```
location /status {
```

```
stub_status on;
```

```
access_log off;
```

```
}
```

```
.....
```

```
[root@web01 ~]# for ip in 7 8;do curl 172.16.1.$ip/status ;done
```

```
Active connections: 1
```

```
server accepts handled requests
```

```
73 73 69
```

```
Reading: 0 Writing: 1 Waiting: 0
```

```
Active connections: 1
```

```
server accepts handled requests
```

```
134 134 127
```

```
Reading: 0 Writing: 1 Waiting: 0
```

在nginx服务器上添加key

```
cat >/etc/zabbix/zabbix_agentd.d/userparameter_nginx_status.conf <<'EOF'
```

```
UserParameter=nginx_active,curl -s 127.0.0.1/status|awk '/Active/ {print $NF}'
```

```
UserParameter=nginx_accepts,curl -s 127.0.0.1/status|awk 'NR==3 {print $1}'
```

```
UserParameter=nginx_handled,curl -s 127.0.0.1/status|awk 'NR==3 {print $2}'
```

```
UserParameter=nginx_requests,curl -s 127.0.0.1/status|awk 'NR==3 {print $3}'
```

```
UserParameter=nginx_reading,curl -s 127.0.0.1/status|awk 'NR==4 {print $2}'
```

```
UserParameter=nginx_writing,curl -s 127.0.0.1/status|awk 'NR==4 {print $4}'
```

```
UserParameter=nginx_waiting,curl -s 127.0.0.1/status|awk 'NR==4 {print $6}'
```

```
EOF
```

重启Agent服务

服务端测试

```
[root@m01 ~]# zabbix_get -s 172.16.1.7 -p 10050 -k "nginx_waiting"
```

```
0
```

```
[root@m01 ~]# zabbix_get -s 172.16.1.8 -p 10050 -k "nginx_waiting"
```

```
0
```

```
[root@m01 ~]# zabbix_get -s 172.16.1.9 -p 10050 -k "nginx_waiting"
```

```
0
```

在zabbix-web上添加模板



监控项

| Wizard | 名称▲ | 触发器 | 键值 | 间隔 | 历史记录 | 趋势 | 类型 | 应用集 | 状态 |
|--------------------------|------|-----|----------------|----|------|------|------------|--------------|-----|
| <input type="checkbox"/> | 写 | | nginx_writing | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 处理请求 | | nginx_handled | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 接收请求 | | nginx_accepts | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 活动连接 | | nginx_active | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 等待 | | nginx_waiting | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 请求数 | | nginx_requests | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |
| <input type="checkbox"/> | 读 | | nginx_reading | 1s | 90d | 365d | Zabbix 客户端 | nginx status | 已启用 |

@51CTO博客

添加图形

所有模板 / Template Nginx 7 status 应用集 1 监控项 7 触发器 图形 2 聚合图形 自动发现规则 Web 场景

| <input type="checkbox"/> 名称 ▲ | 宽 | 高 | 图形类别 |
|-----------------------------------|-----|-----|------|
| <input type="checkbox"/> Nginx状态 | 900 | 200 | 正常 |
| <input type="checkbox"/> Nginx状态2 | 900 | 200 | 正常 |

@51CTO博客

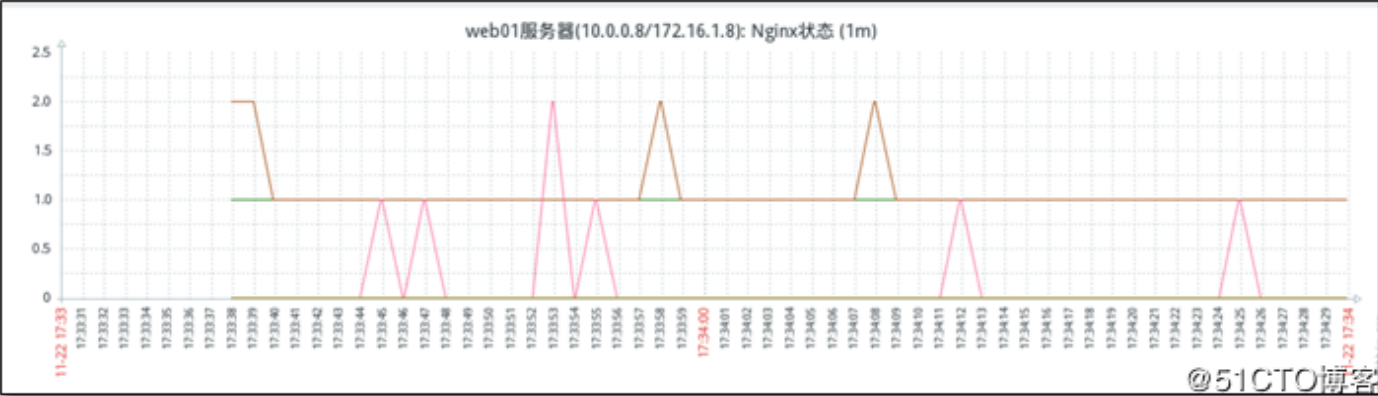
@51CTO博客

将模板关联到主机

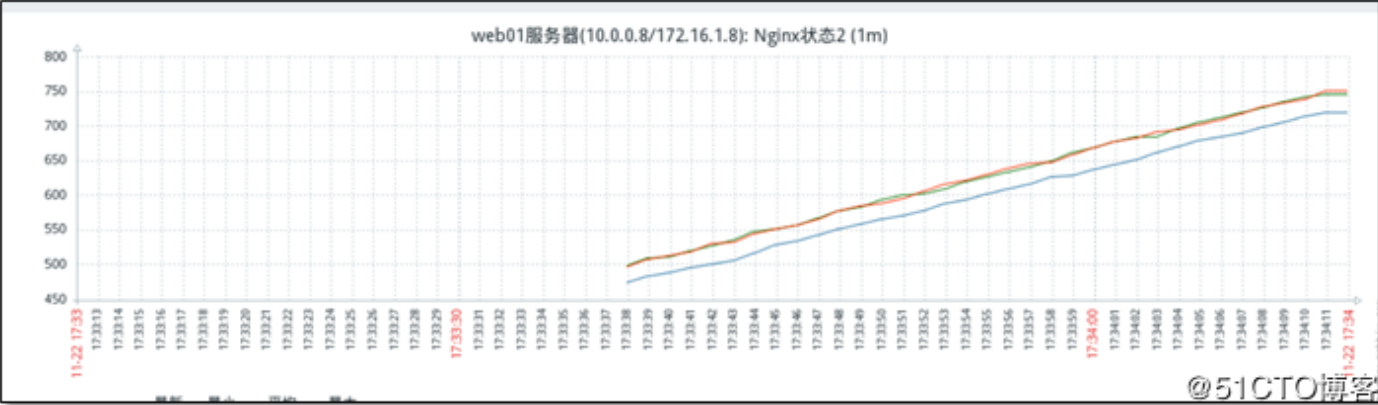
| | | | | | | | | | | | |
|--|--------|--------|--------|-------|--------|---------|------------------|--|--|-----|---------------------|
| <input type="checkbox"/> web03服务器(10.0.0.9/172.16.1.8) | 应用集 13 | 监控项 55 | 触发器 21 | 图形 12 | 自动发现 2 | Web监控 1 | 172.16.1.8:10050 | Template App Nginx Template OS Linux (Template App Zabbix Agent), Template Web Url | Template Nginx 7 status, Template OS Linux (Template App Zabbix Agent), Template Web Url | 已启用 | ZBX SNMP JMX IPMI 无 |
| <input type="checkbox"/> web01服务器(10.0.0.8/172.16.1.8) | 应用集 13 | 监控项 55 | 触发器 21 | 图形 12 | 自动发现 2 | Web监控 1 | 172.16.1.8:10050 | Template App Nginx Template OS Linux (Template App Zabbix Agent), Template Web Url | Template Nginx 7 status, Template OS Linux (Template App Zabbix Agent), Template Web Url | 已启用 | ZBX SNMP JMX IPMI 无 |
| <input type="checkbox"/> web02服务器(10.0.0.7/172.16.1.7) | 应用集 13 | 监控项 55 | 触发器 21 | 图形 12 | 自动发现 2 | Web监控 1 | 172.16.1.7:10050 | Template App Nginx Template OS Linux (Template App Zabbix Agent), Template Web Url | Template Nginx 7 status, Template OS Linux (Template App Zabbix Agent), Template Web Url | 已启用 | ZBX SNMP JMX IPMI 无 |

@51CTO博客

查看添加的图形



@51CTO博客



@51CTO博客

自动发现与自动注册

6.1 自动发现与自动注册

6.1.1 简介

自动发现：

zabbix Server主动发现所有客户端，然后将客户端登记自己的小本本上，缺点zabbix server压力山大（网段大，客户端多），时间消耗多。

自动注册：

zabbix agent主动到zabbix Server上报到，登记；缺点agent有可能找不到Server（配置出错）

6.1.2 两种模式

被动模式：默认 agent被server抓取数据（都是在agent的立场上说）

主动模式：agent主动将数据发到server端（都是在agent的立场上说）

注意：两种模式都是在**agent**上进行配置

zabbix 的使用要在**hosts**文件中预先做好主机名的解析

6.2 自动发现--被动模式

自动发现（被动模式）

第一步：zabbix Server安装完毕（完成）

第二步：zabbix agent安装完毕，Server=172.16.1.61（完毕）

第三步：网页上配置自动发现规则

@51CTO博客

第一个里程碑：完成之前的安装

注意：在Server端做好host解析

zabbix Server安装完毕

第二个里程碑：配置agent客户端

zabbix agent安装完毕，注意配置Server=172.16.1.61

第三个里程碑：在web界面上进行配置

web界面：配置 >> 自动发现 >> Local network

使用自带的自动发现规则（进行修改）即可



在ip范围内输入ip，注意格式；

延迟在实际的生产环境中要大一些（1h-2h），实验环境可以小一些60s

自动发现规则

名称

由agent代理程序自动发现

IP范围

延迟(秒) 测试

检查 [编辑](#) [移除](#)
[新的](#)

设备唯一性准则
☒ IP地址
☐ Zabbix 客户端 "system.username"

已启用 ☒

@51CTO博客

创建发现动作

配置 >> 动作 >> Auto discovery. Linux servers.

ZABBIX 配置

主机群组 模板 主机 维护 动作 自动发现 IT服务

动作

| 名称 | 条件 | 操作 | 状态 |
|-------------------------------|--|--|-----|
| Auto discovery Linux servers. | 接收到的值 似 Linux 自动发现状态 = 启动 服务类型 = Zabbix 客户端 主机IP地址 = 172.16.1.1-254 | 添加主机 添加到主机群组: Linux servers 替换到模板: Template OS Linux 启用主机 | 已启用 |

显示已自动发现的 1 中的 1

@51CTO博客

① 配置动作

动作

动作 条件 操作

名称

默认接收人

默认信息

已启用 ☒

@51CTO博客

② 在条件中添加条件，让添加更准确（条件保持不变，D不需要添加）

动作

动作

条件

操作

计算方式

与/或

A and B and C and D

| 条件 | 标签 | 名称 | 动作 |
|----|----|-------------------------|--------------------|
| A | | 接收到的值 似 Linux | 移除 |
| B | | 自动发现状态 = 启动 | 移除 |
| C | | 服务类型 = Zabbix 客户端 | 移除 |
| D | | 主机IP地址 = 172.16.1.1-254 | 移除 |

新的触发条件

接收到的值

似

[添加](#)

更新

克隆

删除

取消

@51CTO博客

- ③ 在操作中添加
- a) 添加主机与启用主机

动作

动作

条件

操作

动作操作

细节

动作

添加主机

添加到主机群组: Linux servers

链接到模板: Template OS Linux

启用主机

新的

更新

克隆

删除

取消

@51CTO博客

然后等待客户端自动上门就好

小伙伴们可以关注我的微信公众号：linux运维菜鸟之旅



关注“中国电信天津网厅”公众号，首次绑定可免费领2G流量，为你的学习提供流量！



版权声明：原创作品，如需转载，请注明出处。否则将追究法律责任
