**老男孩linux运维实战培训**

**老男孩教育教学核心思想6重：重目标、重思路、重方法、重实践、重习惯、重总结**

**学无止境，老男孩教育成就你人生的起点！**

**版权声明：**

本文作者为《老男孩linux运维实战培训》学生—**你的名字**

本文的所有内容均来自老男孩培训**命令总结**，未经本人及老男孩培训许可，禁止私自转发及使用。

QQ: **你的**XXXX

E-mail: **你的**XXXX

**联系方式:**

|  |  |  |  |
| --- | --- | --- | --- |
| 网站运维交流群： | | | |
| **Linux QQ交流群 385168604** | | **架构师 QQ交流群 390642196** | |
| **Python QQ交流群 29215534** | | **大数据 QQ交流群 421358633** | |
| =================================================================================== | | | |
| **老男孩linux实训联系方式** |  | |  |
| **咨询QQ** | **41117397（丹丹）** | | **70271111（歪歪）** |
|  | **80042789（飞雪）** | | **390320151（小雨）** |
|  | **41117483（冰冰）** | |  |
| **电话：** | **158-1059-0206（丹丹）** | | **189-1171-8229（歪歪）** |
|  | **135-5261-2571（飞雪）** | | **186-0046-2391（小雨）** |
| **网站:** | <http://www.etiantian.org> | | <http://www.oldboyedu.com> |
| **博客:** | http://oldboy.blog.51cto.com | | <http://blog.oldboyedu.com> |

**快捷方式说明:**

**ctrl + 1 一级标题**

**ctrl + 2 二级标题**

**ctrl + 3 三级标题**

**ctrl + 5 程序代码**

**ctrl + 6 正文**

**格式约定：**

蓝色字体：内容注释

目 录

[Zabbix监控实战总结 1](#_Toc514520019)

[第1章 Zabbix监控nginx 1](#_Toc514520020)

[1.1 nginx主配置文件 1](#_Toc514520021)

[1.2 二级配置文件 2](#_Toc514520022)

[1.2.1 给脚本添加权限 4](#_Toc514520023)

[1.2.2 修改agentd端配置文件 4](#_Toc514520024)

[1.2.3 Zabbix\_Server端测试 5](#_Toc514520025)

[1.3 页面设置主机及控制监控项 5](#_Toc514520026)

[1.3.1 创建应用集nginx\_status 5](#_Toc514520027)

[1.3.2 创建监控项 7项 5](#_Toc514520028)

[1.3.3 监控效果 10](#_Toc514520029)

Zabbix监控实战总结

# Zabbix监控nginx

实验环境

|  |  |  |  |
| --- | --- | --- | --- |
| 服务器系统 | 角色 | IP | ServerName |
| CentOS release 6.7 (Final) | Zabbix-Server | 10.0.0.71 | Zabbix |
| CentOS release 6.7 (Final) | Zabbix-Server | 10.0.0.52 | Mysql |
| CentOS release 6.7 (Final) | Zabbix-Server | 10.0.0.8 | Nginx-web |

## nginx主配置文件

[[root@web01 conf]# pwd

/application/nginx/conf

[root@web01 conf]# cat -n nginx.conf

1 error\_log logs/error.log error;

2 worker\_processes 1;

3 events {

4 worker\_connections 1024;

5 }

6 http {

7 include mime.types;

8 default\_type application/octet-stream;

9 sendfile on;

10 keepalive\_timeout 65;

11 log\_format main '$http\_x\_forwarded\_for - $remote\_user [$time\_local] "$request" '

12 '$status $body\_bytes\_sent "$http\_referer" '

13 '"$http\_user\_agent" "$remote\_addr"';

14 #include extra/index.conf;

15 include extra/nginx\_status.conf;

16 include extra/www.conf;

17 include extra/blog.conf;

18 include extra/bbs.conf;

19 }

## 二级配置文件

/application/nginx/conf/extra

[root@web01 extra]# cat -n nginx\_status.conf

1 server{

2 location /nginx\_status {

3 stub\_status on;

4 access\_log off;

5 allow 127.0.0.1;

6 deny all;

7 }

8 }

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[root@web01 etc]# curl -s http://127.0.0.1:80/nginx\_status

Active connections: 1

server accepts handled requests

117 117 140

Reading: 0 Writing: 1 Waiting: 0

Nginx状态解释：

Active connections Nginx正处理的活动链接数1个

server Nginx启动到现在共处理了1个连接。

accepts Nginx启动到现在共成功创建1次握手。

handled requests Nginx总共处理了1次请求。

Reading Nginx读取到客户端的 Header 信息数。

Writing Nginx返回给客户端的 Header 信息数。

Waiting Nginx已经处理完正在等候下一次请求指令的驻留链接，开启。

Keep-alive的情况下，这个值等于active-（reading + writing）。

请求丢失数=(握手数-连接数)可以看出,本次状态显示没有丢失请求。

编写Nginx的Shell脚本(如果端口不一致,只需要修改脚本端口即可)

[root@web01 extra]# cd /etc/zabbix/etc/zabbix\_agentd.conf.d/scripts/

[root@web01 scripts]# cat nginx/nginx\_status.sh

#!/bin/bash

#

#

#

NGINX\_PORT=80

NGINX\_COMMAND=$1

nginx\_active(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk '/Active/ {print $NF}'

}

nginx\_reading(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk '/Reading/ {print $2}'

}

nginx\_writing(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk '/Writing/ {print $4}'

}

nginx\_waiting(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk '/Waiting/ {print $6}'

}

nginx\_accepts(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk 'NR==3 {print $1}'

}

nginx\_handled(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk 'NR==3 {print $2}'

}

nginx\_requests(){

/usr/bin/curl -s "http://127.0.0.1:"$NGINX\_PORT"/nginx\_status/" |awk 'NR==3 {print $3}'

}

case $NGINX\_COMMAND in

active)

nginx\_active;

;;

reading)

nginx\_reading;

;;

writing)

nginx\_writing;

;;

waiting)

nginx\_waiting;

;;

accepts)

nginx\_accepts;

;;

handled)

nginx\_handled;

;;

requests)

nginx\_requests;

;;

\*)

echo $"USAGE:$0 {active|reading|writing|waiting|accepts|handled|requests}"

esac

### 给脚本添加权限

[root@web01 scripts]# chmod +x nginx\_status.sh

### 修改agentd端配置文件

[root@web01 etc]# egrep -v "^#|^$" zabbix\_agentd.conf

LogFile=/tmp/zabbix\_agentd.log

Server=172.16.1.71

ServerActive=172.16.1.71

Hostname=web01

Timeout=8

UnsafeUserParameters=1

UserParameter=nginx[\*],/etc/zabbix/etc/zabbix\_agentd.conf.d/scripts/nginx/nginx\_status.sh "$1" 🡨监控项脚本存放路径

UserParameter=login-user,who|wc -l 🡨监控用户数量

### Zabbix\_Server端测试

root@zabbix zabbix]# /application/zabbix/bin/zabbix\_get -s 172.16.1.8 -k nginx

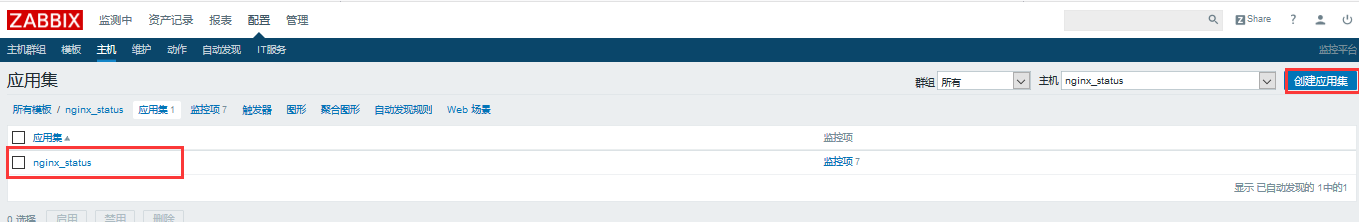
USAGE:/etc/zabbix/etc/zabbix\_agentd.conf.d/scripts/nginx/nginx\_status.sh {active|reading|writing|waiting|accepts|handled|requests}

[root@zabbix zabbix]# /application/zabbix/bin/zabbix\_get -s 172.16.1.8 -k nginx[handled]

116

## 页面设置主机及控制监控项

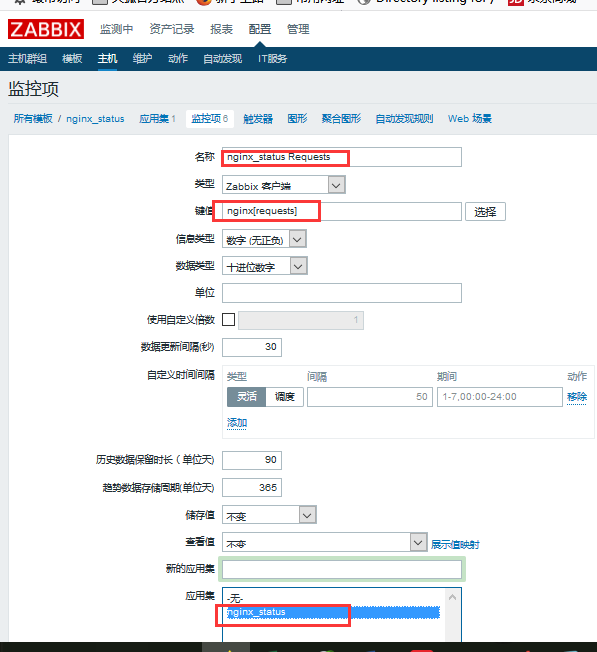
### 创建应用集nginx\_status

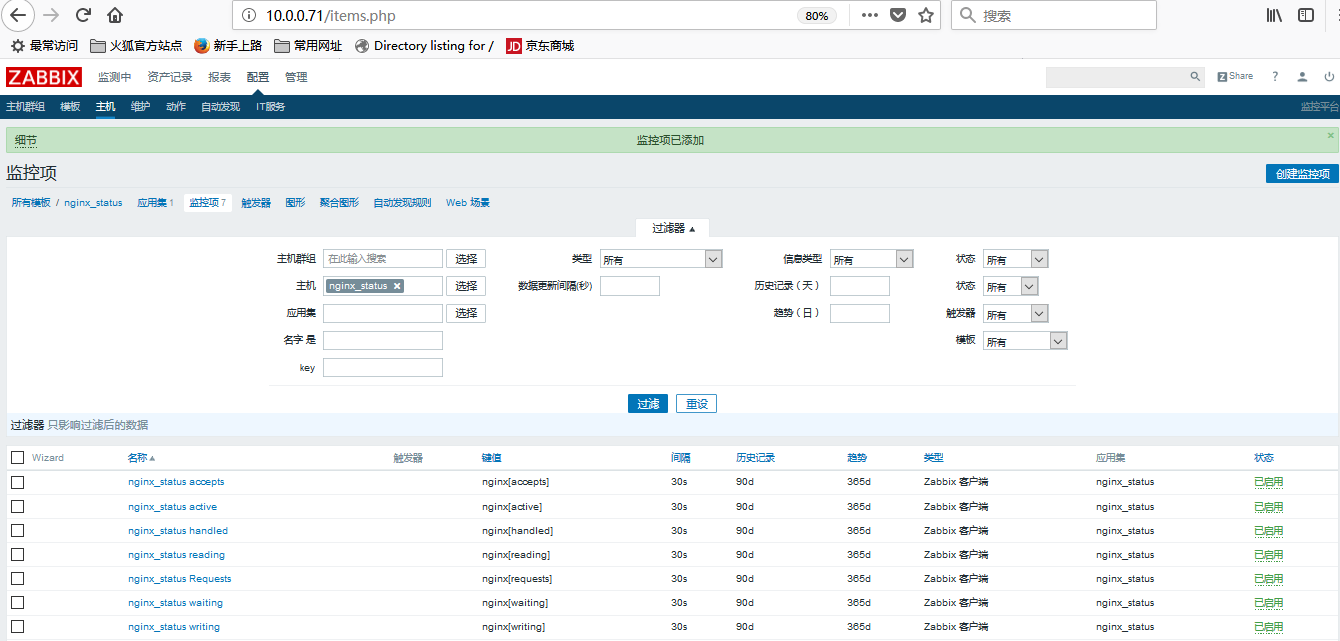


### 创建监控项 7项

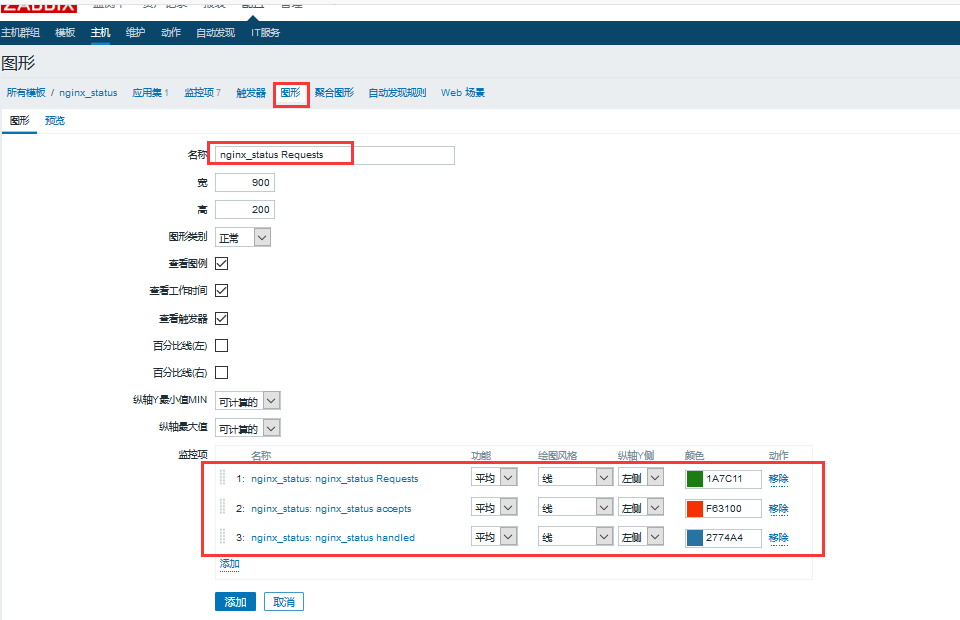
创建item

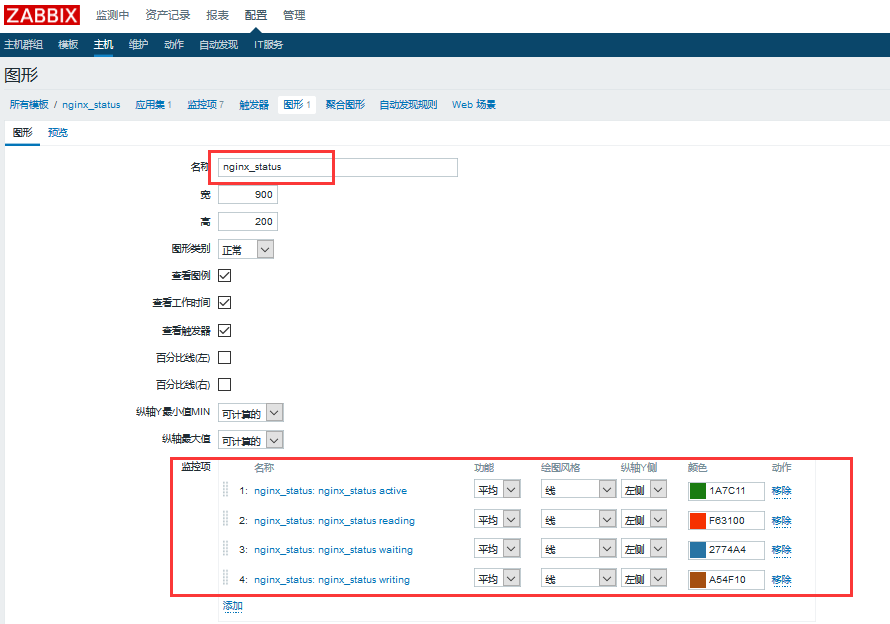
根据zabbix-agent中的nginx\_status.sh的参数，总共要创建active、reading、writing、waiting、accepts、handled、requests这7项item，下面以创建accepts为例，其他创建



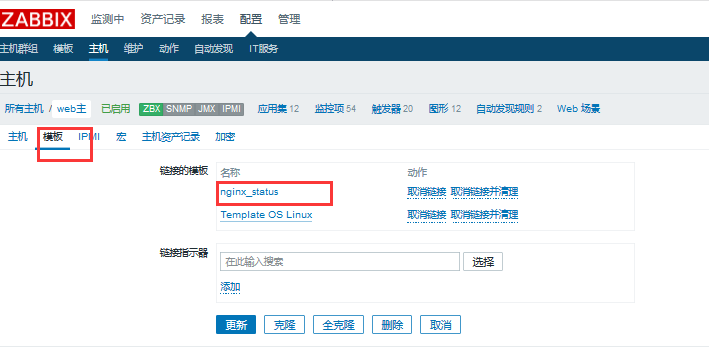


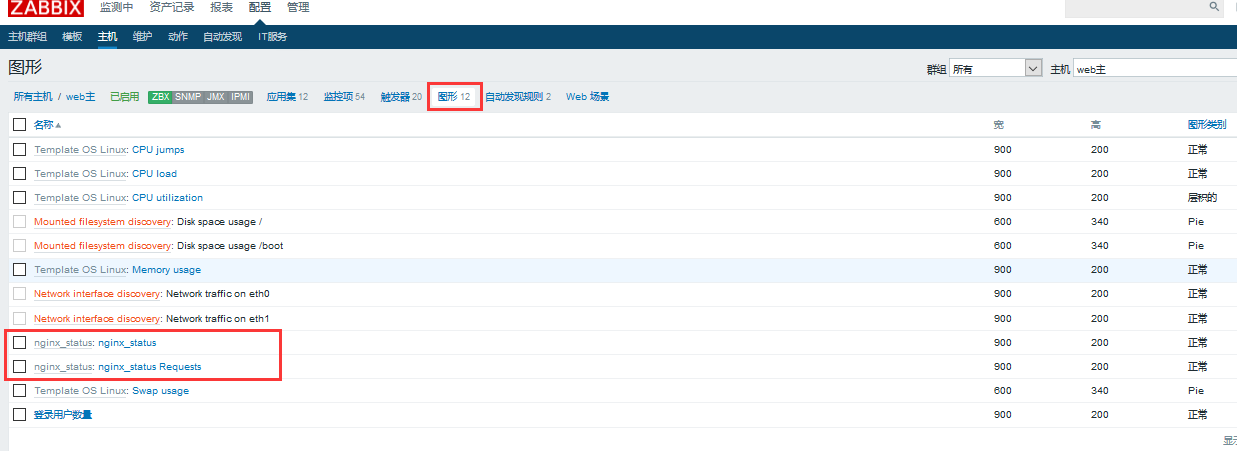
item创建完成后，我们来创建graph 图形，根据参数的不同，创建了两个graphs



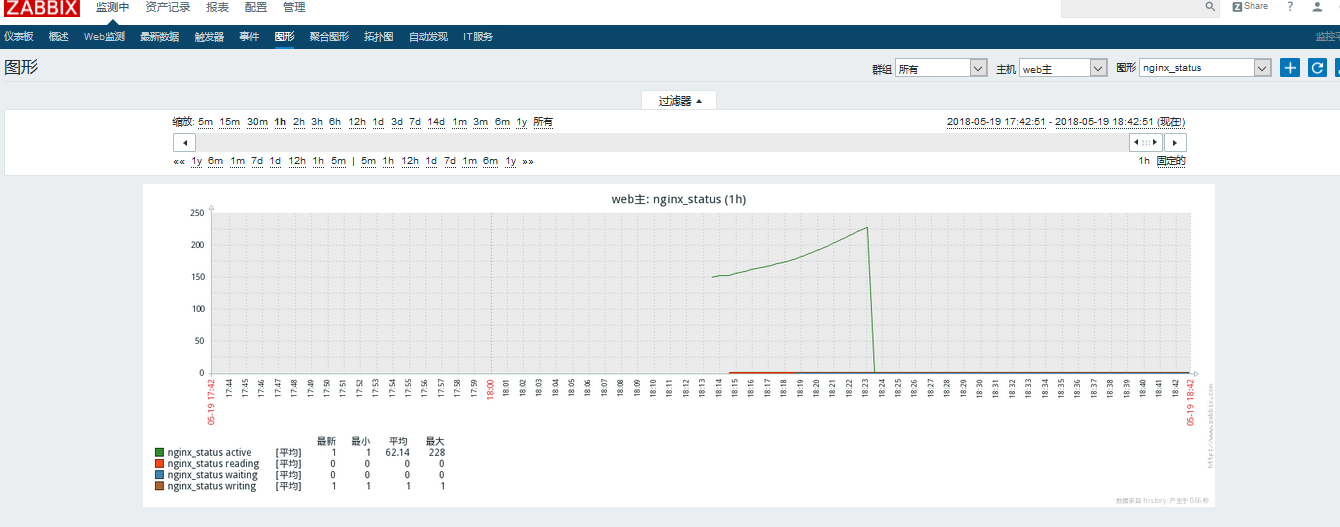


以上就完成的NGINX STATUS MONTIOR Template的创建，下面将该Template添加至zabbix-agent中





### 监控效果





模板导出为nginx\_status\_templates

### Zabbix监控nginx状态总结-思路

1. 利用nginx自带的status模块，配置状态模块
2. Web端测试curl 模块运行状态 curl -s http:127.0.0.1/nginx\_status
3. Zabbix-agentd端scripts目录下编写状态监控脚本nginx\_status.sh,以及修改agentd.conf配置文件
4. Zabbix-server端利用zabbix-get 工具检测server能否正确获取nginx状态
5. Zabbix页面添加nginx主机,模板；应用集；监控项（从stub\_status模块获取的数据）；图形、触发器等
6. 验证监控状态