

Zabbix 自动发现监控 http 端口

版本 V1.0

时间 2012-12-26

版权 GPL

作者 itnihao 邮箱 itnihao@qq.com

博客 <http://itnihao.blog.51cto.com>

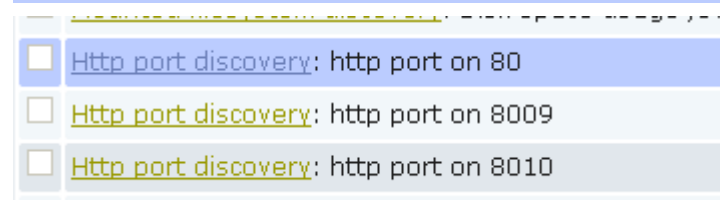
如需重新发行，请注明以上信息，谢谢合作

说明，此文档为简单的说明，是 zabbix 插件自定义的高级使用方法，需要有一定的使用经验。

环境

http 上面有多个端口，基于端口的虚拟主机，每次添加监控 http 端口监控很麻烦，机器很多，手动添加很麻烦，如是研究了一下自动发现功能，特写此文档记录。

Zabbix 2.0 版本提供了 `vfs.fs.discovery` `net.if.discovery` 两个自动发现 key，功能异常强大。经过研究官方文档，终于找到定义自动发现规则的方法。实现后的效果如下



当服务器开启端口的时候，自动发现，自动添加，自动监控，而不需要手工去添加端口项的监控

客户端的配置文件如下。

```
cat /etc/zabbix/zabbix_agentd.conf|grep -v "^#"|grep -v "^$"  
LogFile=/var/log/zabbix/zabbix_agentd.log  
EnableRemoteCommands=1  
Server=127.0.0.1  
ServerActive=127.0.0.1:20051  
Hostname=zabbix.test.com  
Include=/etc/zabbix/zabbix_agentd.conf.d/  
UnsafeUserParameters=1
```

一 http 数据获取脚本的编写。

```
#!/bin/bash  
discovery () {  
netstat -nlput|awk -F":" '{print $0}' |awk -F: '{print $2}'|awk '{print $1}'  
>/var/log/zabbix/http.log  
port=$(cat /var/log/zabbix/http.log)  
printf '\n'  
printf '\t"data":['  
for((i=0;i<${#port[@]};i++))  
{  
printf "\t\t{ \n"
```

```

        printf "\t\t\t" {#APACHEPORT} \:"$ {port[$i]} \"},\n"
#       printf "\t\t\t" {#apache_name} \:"http_$ {port[$i]} \"},\n"
        num=$(echo ${#port[@]}-1)
        if [ "$i" == $num ];
            then
                printf "\t\t\t\n"
                printf "\t\t\t" {#APACHEPORT} \:"$ {port[$num]} \"}\n"
#               printf "\t\t\t" {#apache_name} \:"http_$ {port[$num]} \"}\n"
            fi
        }
    }

http_port () {
PORT=$(curl -o /dev/null -s -w %{http_code} http://127.0.0.1:$1/)
[ $PORT == 200 ] && echo ${PORT}
[ $PORT != 200 ] && echo ${PORT}
}

case "$1" in
port)
http_port $2
;;
discovery)
discovery
;;
test)
echo test
;;
esac

```

```
shell#chown    zabbix .zabbix    /var/log/zabbix/http.log
```

```

pts]# ll /etc/zabbix/scripts/http
root root 837 12月 25 18:43 /etc/zabbix/scripts/http
pts]#

```

```

cat /etc/zabbix/zabbix_agentd.conf.d/http.conf
UserParameter=vfs.http.discovery,/etc/zabbix/scripts/http discovery
UserParameter=vfs.http.port[*],/etc/zabbix/scripts/http port $1
UserParameter=vfs.http.test,/etc/zabbix/scripts/http test

```

```
[root@nat scripts]# cat /etc/zabbix/zabbix_agentd.conf.d/http.conf
UserParameter=vfs.http.discovery,/etc/zabbix/scripts/http_discovery
UserParameter=vfs.http.port[*],/etc/zabbix/scripts/http_port $1
UserParameter=vfs.http.test,/etc/zabbix/scripts/http_test
[root@nat scripts]#
```

```
[root@nat scripts]# zabbix_get -s 127.0.0.1 -k vfs.http.discovery
{
  "data": [
    {
      "{#APACHEPORT}": "8009",
    },
    {
      "{#APACHEPORT}": "8010",
    },
    {
      "{#APACHEPORT}": "80",
    },
    {
      "{#APACHEPORT}": "80"
    }
  ]
}
```

```
[root@nat scripts]# zabbix_get -s 127.0.0.1 -k vfs.http.port[80]
200
[root@nat scripts]# zabbix_get -s 127.0.0.1 -k vfs.http.port[8009]
403
[root@nat scripts]# zabbix_get -s 127.0.0.1 -k vfs.http.port[8010]
403
[root@nat scripts]#
```

OK,至此,zabbix 的自定义脚本获取数据已经实现

二,模板的添加

定义全局的正则表达式

ZABBIX

状态统计 | 资产记录 | 系统评估 | 系统配置 | **管理**

总览 | 节点管理 | 认证 | 用户 | 处理方法 | 脚本 | 审计 | 队列 | 警报 | 安装

访问记录: 最新报警 » Custom graphs » 最新报警 » Configuration of proxies » Configuration of GUI

正则表达式配置

正则表达式

<input type="checkbox"/> 名称	表达式
<input type="checkbox"/> Apache	1 & ^(/018000018010)\t (Result is TRUE)

正则表达式

GUI
Housekeeper
图片
Icon mapping
正则表达式
Macros
Value mapping
工作时间
Trigger severities
Trigger displaying options
其他

正则表达式

名称 Apache 名称

Test string

8009

表达式

表达式	期望的结果	Case sensitive	编辑
<input type="checkbox"/> ^ (80 8009 8010)\$	Result is TRUE	是	编辑

New expression

表达式 ^ (80|8009|8010)\$

Expression type Result is TRUE

Case sensitive ☒

保存 取消

注意，此处的正则表达式是只监控 80，8009，8010 这 3 个端口，如果是这 3 个端口，结果就为真，也可以定义结果为 FALSE，表示排除此值。

正则表达式配置

正则表达式

名称	表达式
Apache	1 » ^ (80 8009 8010)\$ [Result is TRUE]

下面开始在主机里面添加自动发现 http 端口，此处演示的是单个主机添加，如果要作为模板，需要添加到模板，然后应用到主机，方法是相同的。

The screenshot shows the Zabbix web interface. At the top, the 'System Configuration' (系统配置) tab is selected. Below it, the 'Hosts' (主机) tab is selected. The breadcrumb trail is 'Configuration of proxies » Configuration of hosts » Latest alerts »'. The 'Host Configuration' (主机配置) section is active, and the 'Hosts' (主机) sub-section is selected. A red box highlights the 'Hosts' tab. Below the breadcrumb trail, there is a 'Create discovery rule' button. A table shows the number of items for each category: Application sets (12), Monitoring items (93), Triggers (26), Graphs (23), and Automatic discovery (3). The 'Automatic discovery' link is circled in red. Below the table, there is a 'Create discovery rule' button. The 'Create discovery rule' form is shown with the following fields: Name (名称): 'Http port discovery' (circled in red); Type (类型): 'Zabbix agent (被动)'; Key (Key): 'vfs.http.discovery' (circled in red); Host interface: '10050'; Data update interval (秒): '30'; Flexible intervals: A table with columns 'Interval', 'Period', and 'Action'. The 'Interval' column has a value of '50', the 'Period' column has a value of '1-7,00:00-24:00', and the 'Action' column has a value of '添加'. The 'New flexible interval' section shows 'Interval (in sec)' as '50' and 'Period' as '1-7,00:00-24:00'. The 'Filter' section shows 'Macro' as '{ #APACHEPOF' and 'Regexp' as '@Apache' (both circled in red). The 'Detection content' (检测内容) field is empty. The 'Status' (状态) dropdown is set to 'Active' (活跃). At the bottom, there are buttons for 'Save' (保存), 'Copy' (复制), 'Delete' (删除), and 'Cancel' (取消).

名称: Http port discovery

类型: Zabbix agent (被动)

Key: vfs.http.discovery

Host interface: 10050

数据更新间隔(秒): 30

Flexible intervals

间隔	时期	操作
No flexible intervals defined.		

New flexible interval

Interval (in sec)	时期	操作
50	1-7,00:00-24:00	添加

st resources period (in days): 30

Filter

Macro: { #APACHEPOF

Regexp: @Apache

检测内容

状态: 活跃

保存 复制 删除 取消

名称, key 值为前面脚本定义的, 即

UserParameter=vfs.http.discovery,/etc/zabbix/scripts/http discovery

<input type="checkbox"/> 名称	监控项	触发器
<input type="checkbox"/> Http port discovery	Item prototypes (1)	Trigger pr

点击 Item，创建监控项

[Create item prototype](#)

名称	<input type="text" value="http port on \$1"/>								
类型	<input type="text" value="Zabbix agent (被动)"/>								
Key_	<input type="text" value="vfs.http.port[{ #APACHEPORT}]"/>		<input type="button" value="搜索"/>						
Host interface	<input type="text" value="10.10.10.10: 10050"/>								
数据类型	<input type="text" value="Numeric (unsigned)"/>								
Data type	<input type="text" value="Decimal"/>								
单位Units	<input type="text"/>								
Use custom multiplier	<input type="checkbox"/> <input type="text" value="1"/>								
数据更新间隔(秒)	<input type="text" value="30"/>								
Flexible intervals	<table><thead><tr><th>间隔</th><th>时期</th><th>操作</th></tr></thead><tbody><tr><td colspan="3">No flexible intervals defined.</td></tr></tbody></table>			间隔	时期	操作	No flexible intervals defined.		
间隔	时期	操作							
No flexible intervals defined.									
New flexible interval	<table><tr><td>Interval (in sec)</td><td><input type="text" value="50"/></td><td>时期</td><td><input type="text" value="1-7,00:00-24:00"/></td><td><input type="button" value="添加"/></td></tr></table>			Interval (in sec)	<input type="text" value="50"/>	时期	<input type="text" value="1-7,00:00-24:00"/>	<input type="button" value="添加"/>	
Interval (in sec)	<input type="text" value="50"/>	时期	<input type="text" value="1-7,00:00-24:00"/>	<input type="button" value="添加"/>					
数据保存天数	<input type="text" value="7"/>								
Keep trends (in days)	<input type="text" value="365"/>								
Store value	<input type="text" value="不变"/>								
Show value	<input type="text" value="不变"/> 展示值对应								
New application	<input type="text"/>								
应用集	<div><div>-None-</div><div>CPU</div><div>Filesystems</div><div>General</div><div>Memory</div><div>MySQL</div></div>								
检测内容	<div>http端口自动发现</div>								
活跃	<input checked="" type="checkbox"/>								

名称这里应用的是端口变量，为\$1，
Key 值这里是变量引用
vfs.http.port[{#APACHEPORT}]
{#APACHEPORT}这个是 vfs.http.discovery 中的量

创建图像

图形显示
Graph prototypes (1)

Create graph prototype

Preview

名称: http port on {#APACHEPORT}

宽: 900
高: 200

Graph type: Normal

Show legend: ☒
Show working time: ☒
Show triggers: ☒
Percentile line (left): ☐
Percentile line (right): ☐

纵轴最小值: 计算
纵轴最大值: 计算

名称	Function	Draw style	纵轴	颜色	操作
1: http port on {#APACHEPORT}	平均	Line	左侧	00C800	Remove
2: http port on {#APACHEPORT}	平均	Line	左侧	EE00EE	Remove

添加 添加原型

保存 复制 删除 取消

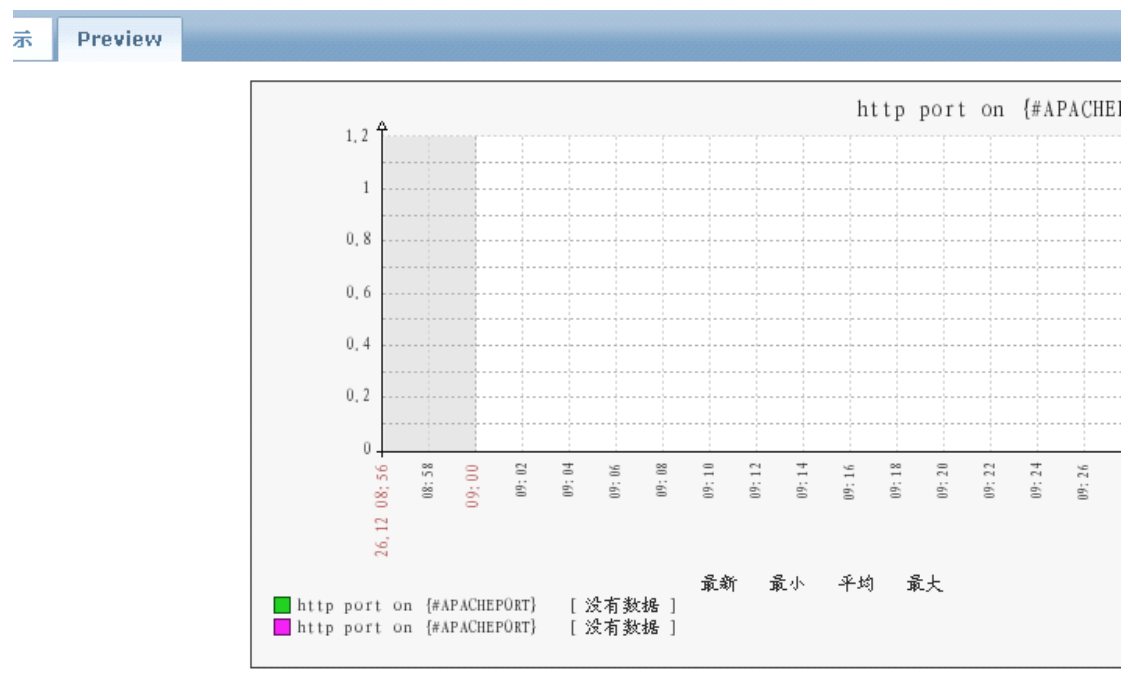
注意，名称依然是变量引用

名称	Key
{#APACHEPORT}	{#APACHEPORT}
http port on {#APACHEPORT}	vfs.http.port[{#APACHEPORT}]

添加 添加原型

搜索

选择添加原型，预览一下图像



最终添加的自动发现规则如下

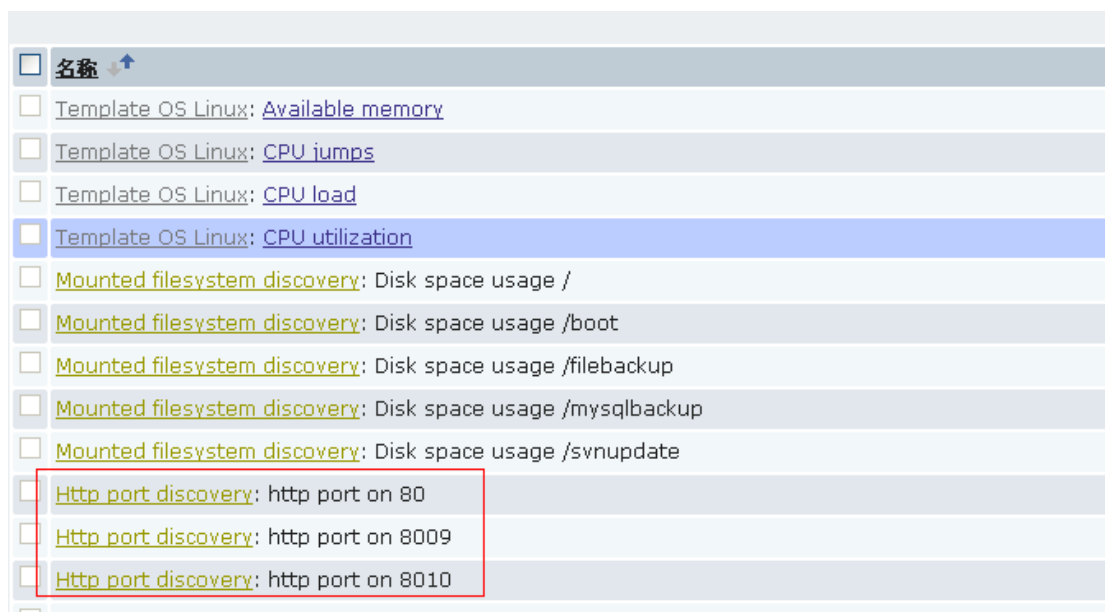
名称	监控项	触发器	图形显示	Key	间隔	类型	状态	错误
http port discovery	Item prototypes (1)	Trigger prototypes (0)	Graph prototypes (1)	vfs.http.discovery	30	Zabbix agent (被动)	活跃	

30 秒后，自动发现，可以看到如下效果

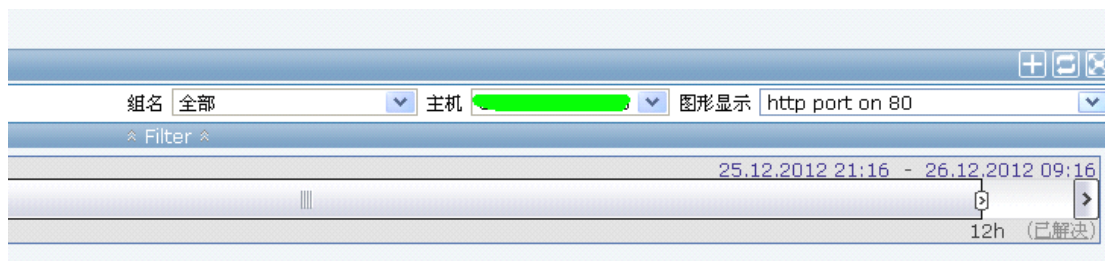
监控项 (93)

									Inte
<input checked="" type="checkbox"/>	Http port discovery : http port on 8009		vfs.http.port[8009]	30	7	365	Zabbix agent (被动)		
<input type="checkbox"/>	Http port discovery : http port on 80		vfs.http.port[80]	30	7	365	Zabbix agent (被动)		
<input type="checkbox"/>	Http port discovery : http port on 8010		vfs.http.port[8010]	30	7	365	Zabbix agent (被动)		

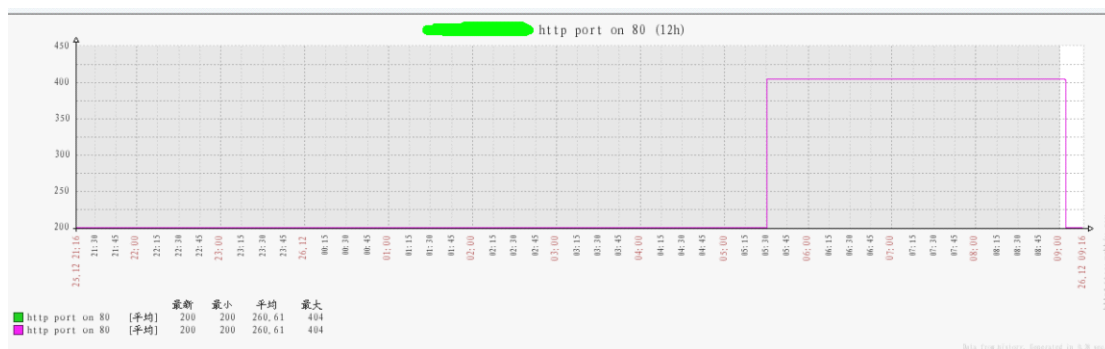
图形显示 (23)



点击 **图形显示**，选择主机，选择图形显示



可以看到图像自动添加上去了





至此，http 自动发现端口已经实现，对于监控其他应用，举一反三即可