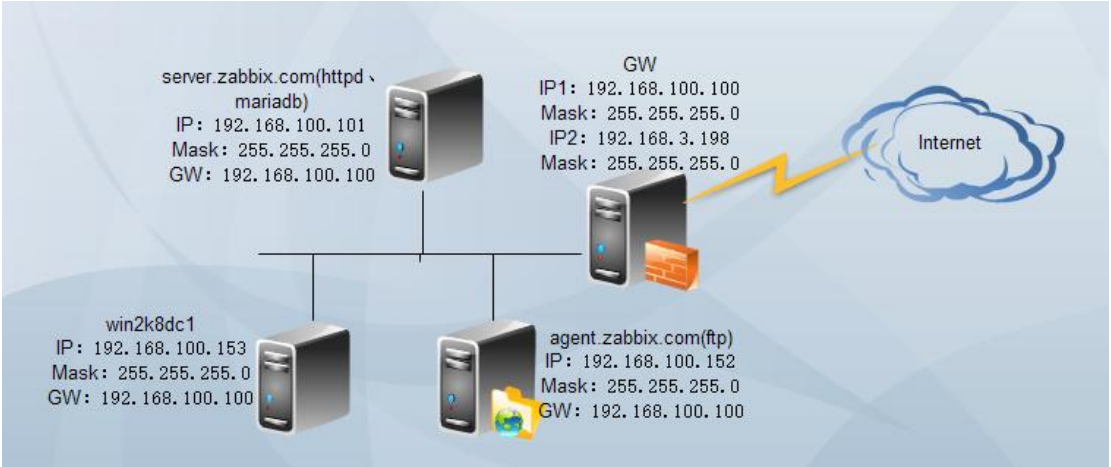


第五十四章：Zabbix 监控结合 Grafana 绘图

一.项目实验拓扑图（使用亿图专家绘图）;



二.项目实验思路及重点内容(步骤及重难点知识);

环境:

主机	操作系统	ip 地址	主要软件
server.zabbix.com	Centos 7.4.1708 64 bit	192.168.100.101/24	Lamp 、 zabbix-3.4.2.tar.gz 、 grafana-4.6.1-1.x86_64.rpm、 mailx、
agent.zabbix.com	Centos 6.5 64 bit	192.168.100.152/24	zabbix-3.4.2.tar.gz 、 vsftpd
Win2k8dc1	Windows 2008 R2 64 bit	192.168.100.153/24	zabbix_agents_3.4.0. win

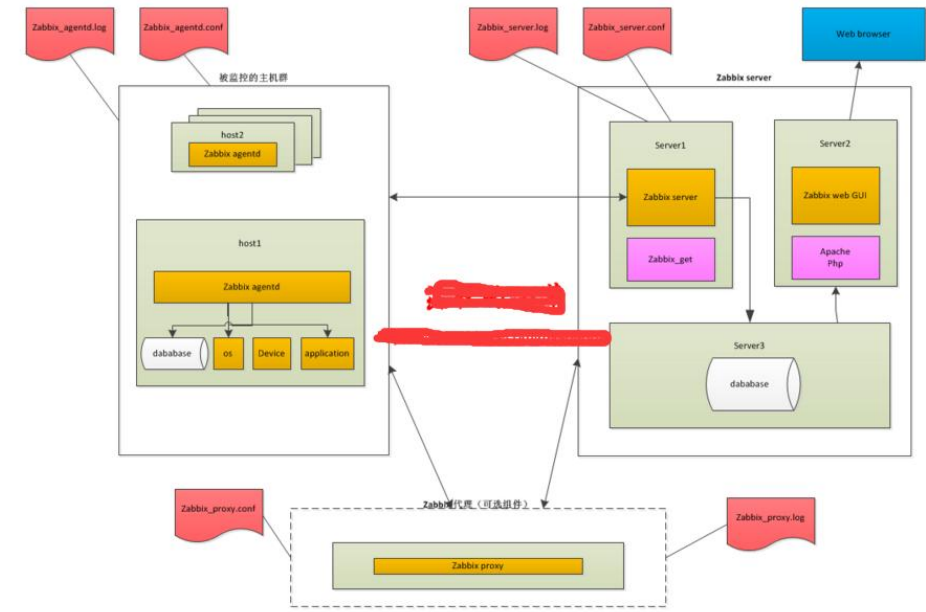
实施步骤:

步骤	说明
第一步	1.安装 zabbix server 服务端;
第二步	2.配置 zabbix 服务的 web 页面并做简单优化;
第三步	3.实现监控 server.zabbix.com 端的 OS 以及 mysql、httpd;

第四步	4.安装及配置 agent.zabbix.com 被监控端（FTP 服务器）；
第五步	5.登录 zabbix 的 web 界面，进行添加 agent 主机，验证监控效果；
第六步	6.配置监控 Win2k8dc1 主机，验证效果；
第七步	7.自定义监控项，创建项目、触发器、图形，验证监控效果；
第八步	8.实现邮件报警；
第九步	9.安装并配置 Grafana；
第十步	10.配置 Grafana 的 web 界面，呈现效果；

重点、难点：

重点一	<p>1. Zabbix 简介</p> <p>Zabbix 是一个企业级的开源分布式监控解决方案，由 C 语言编写而成的底层架构（server 端和 agent 端），由一个国外的团队持续维护更新，软件可以自由下载使用，运作团队靠提供收费的技术支持赢利。</p> <p>官方网站：http://www.zabbix.com</p> <p>Zabbix 通过 C/S 模式采集数据，通过 B/S 模式在 web 端展示和配置。</p> <p>被监控端：主机通过安装 agent 方式采集数据，网络设备通过 SNMP 方式采集数据</p> <p>Server 端：通过收集 SNMP 和 agent 发送的数据，写入 MySQL 数据库，再通过 php+apache 在 web 前端展示。</p> <p>2. Zabbix 运行条件：</p> <p>Server：</p> <p>Zabbix Server 需运行在 LAMP（Linux+Apache+Mysql+PHP）环境下，对硬件要求低</p> <p>Agent：</p> <p>目前已有的 agent 基本支持市面常见的 OS，包含 Linux、HPUX、Solaris、Sun、windows 等</p> <p>SNMP：</p> <p>支持各类常见的网络设备</p> <p>3. Zabbix 功能</p> <p>具备常见的商业监控软件所具备的功能（主机的性能监控、网络设备性能监控、数据库性能监控、FTP 等通用协议监控、多种告警方式、详细的报表图表绘制）支持自动发现网络设备和服务器，支持分布式，能集中展示、管理分布式的监控点，扩展性强，server 提供通用接口，可以自己开发完善各类监控。</p>
重点二	<p>1. 优劣势</p> <p>优点：</p> <p>开源，无软件成本投入；</p> <p>Server 对设备性能要求低（实际测试环境：虚拟机 Redhat EL AS5，2G CPU 1G 内存，监控 5 台设备，CPU 使用率基本保持在 10%以下，内存剩余 400M 以上）；</p> <p>支持设备多；</p> <p>支持分布式集中管理；</p> <p>开放式接口，扩展性强；</p> <p>当监控的 item 比较多服务器队列比较大时可以采用被动状态，被监控客户端主动从 server 端去下载需要监控的 item 然后取数据上传到 server 端。这种方式</p>

	<p>对服务器的负载比较小。</p> <p>缺点：</p> <p>无厂家支持，出现问题解决比较麻烦</p> <p>需在被监控主机上安装 agent，所有数据都存在数据库里，产生的数据据很大，瓶颈主要在数据库。</p>
重点三	<p>1.zabbix 的监控原理：</p>  <p>The diagram illustrates the Zabbix monitoring architecture. On the left, a group of monitored hosts (被监控的主机群) includes host1 and host2. Host1 contains a Zabbix agentd component that interacts with the OS, Device, and application layers, and stores data in a database. Host2 also contains a Zabbix agentd component. On the right, the Zabbix server infrastructure is shown, consisting of Server1, Server2, and Server3. Server1 contains the Zabbix server component and the Zabbix_get component. Server2 contains the Zabbix web GUI and the Apache Php component. Server3 contains a database component. A Web browser is connected to the Zabbix web GUI. A Zabbix proxy (可选组件) is shown at the bottom, which can act as a Zabbix agentd or a Zabbix proxy. The diagram shows data flow from the agents to the server, and from the server to the web interface. A red dashed line indicates a network connection between the hosts and the server.</p> <p>组件说明：</p> <ol style="list-style-type: none">1) zabbix server: 负责接收 agent 发送的报告信息的核心组件，所有配置、统计数据及操作数据都由它组织进行；2) database storage: 专用于存储所有配置信息，以及由 zabbix 收集的数据；3) web interface: zabbix 的 GUI 接口；4) proxy: 可选组件，常用于监控节点很多的分布式环境中，代理 server 收集部分数据转发到 server，可以减轻 server 的压力；5) agent: 部署在被监控的主机上，负责收集主机本地数据如 cpu、内存、数据库等数据发往 server 端或 proxy 端； <p>监控流程：</p> <p>agentd 需要安装到被监控的主机上，它负责定期收集各项数据，并发送到 zabbix server 端，zabbix server 将数据存储到数据库中，zabbix web 根据数据在前端进行展现和绘图。这里 agentd 收集数据分为主动和被动两种模式：</p> <p>主动: agent 请求 server 获取主动的监控项列表，并主动将监控项内需要检测的数据提交给 server/proxy</p> <p>被动: server 向 agent 请求获取监控项的数据，agent 返回数据。</p> <p>客户端守护进程：</p> <p>此进程收集客户端数据，例如 cpu 负载、内存、硬盘使用情况等。</p> <p>zabbix_get</p> <p>zabbix 工具，单独使用的命令，通常在 server 或者 proxy 端执行获取远程客户端信息的命令。通常用户排错。例如在 server 端获取不到客户端的内存数据，我</p>

	<p>们可以使用 <code>zabbix_get</code> 获取客户端的内容的方式来故障排查。</p> <p>zabbix_sender</p> <p><code>zabbix</code> 工具，用于发送数据给 <code>server</code> 或者 <code>proxy</code>，通常用于耗时比较长的检查。很多检查非常耗时间，导致 <code>zabbix</code> 超时。于是我们在脚本执行完毕之后，使用 <code>sender</code> 主动提交数据。</p> <p>zabbix_server</p> <p><code>zabbix</code> 服务端守护进程。<code>zabbix_agentd</code>、<code>zabbix_get</code>、<code>zabbix_sender</code>、<code>zabbix_proxy</code>、<code>zabbix_java_gateway</code> 的数据最终都是提交到 <code>server</code></p> <p>备注：当然不是数据都是主动提交给 <code>zabbix_server</code>，也有的是 <code>server</code> 主动去取数据。</p> <p>zabbix_proxy</p> <p><code>zabbix</code> 代理守护进程。功能类似 <code>server</code>，唯一不同的是它只是一个中转站，它需要把收集到的数据提交/被提交到 <code>server</code> 里。为什么要用代理？代理是做什么的？卖个关子，请继续关注运维生存时间 <code>zabbix</code> 教程系列。</p> <p>zabbix_java_gateway</p> <p><code>zabbix2.0</code> 之后引入的一个功能。顾名思义：Java 网关，类似 <code>agentd</code>，但是只用于 Java 方面。需要特别注意的是，它只能主动去获取数据，而不能被动获取数据。它的数据最终会给到 <code>server</code> 或者 <code>proxy</code>。</p>
重点四	<p>扩展：zabbix 的监控架构</p> <p>在实际监控架构中，<code>zabbix</code> 根据网络环境、监控规模等 分了三架构：<code>server-client</code>、<code>master-node-client</code>、<code>server-proxy-client</code> 三种。</p> <p>1、server-client 架构</p> <p>也是 <code>zabbix</code> 的最简单的架构，监控机和被监控机之间不经过任何代理，直接由 <code>zabbix server</code> 和 <code>zabbix agentd</code> 之间进行数据交互。适用于网络比较简单，设备比较少的监控环境。</p> <p>2、server-proxy-client 架构</p> <p>其中 <code>proxy</code> 是 <code>server</code>、<code>client</code> 之间沟通的一个桥梁，<code>proxy</code> 本身没有前端，而且其本身并不存放数据，只是将 <code>agentd</code> 发来的数据暂时存放，而后再提交给 <code>server</code>。该架构经常是和 <code>master-node-client</code> 架构做比较的架构，一般适用于跨机房、跨网络的中型网络架构的监控。</p> <p>3、master-node-client 架构</p> <p>该架构是 <code>zabbix</code> 最复杂的监控架构，适用于跨网络、跨机房、设备较多的大型环境。每个 <code>node</code> 同时也是一个 <code>server</code> 端，<code>node</code> 下面可以接 <code>proxy</code>，也可以直接接 <code>client</code>。<code>node</code> 有自己的配置文件和数据库，其要做的是将配置信息和监控数据向 <code>master</code> 同步，<code>master</code> 的故障或损坏对 <code>node</code> 其下架构的完整性。</p>
重点五	<p>Grafana 简介：</p> <p><code>Grafana</code> 是一个可视化面板（Dashboard），有着非常漂亮的图表和布局展示，功能齐全的度量仪表盘和图形编辑器，支持 <code>Graphite</code>、<code>zabbix</code>、<code>InfluxDB</code>、<code>Prometheus</code> 和 <code>OpenTSDB</code> 作为数据源。以 <code>InfluxDB</code>（由 <code>go</code> 语言编写，是一个开源，分布式，时间序列，事件，可度量和无外部依赖的数据库）作为底层数据库；</p> <p><code>Grafana</code> 主要特性：灵活丰富的图形化选项；可以混合多种风格；支持白天和夜间模式；多个数据源。</p>

三.项目实验步骤（操作截图或者操作命令）

1.安装 zabbix server 服务端；

```
[root@server ~]# ip a ##配置网络，确保可以上网
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:4f:c9:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.101/24 brd 192.168.100.255 scope global eth0
        valid_lft forever preferred_lft forever
[root@server ~]# vi /etc/hosts
192.168.100.101 server.zabbix.com
192.168.100.152 agent.zabbix.com
192.168.100.153 win2k8dc1
:wq
[root@server ~]# wget -O /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.aliyun.com/repo/Centos-7.repo
[root@server ~]# yum -y install net-snmp net-snmp-devel libxml2 libxml2-devel libcurl-devel
libevent libevent-devel curl curl-devel mysql-devel snmp perl-DBI php-xml php-bcmath
php-mbstring php-ldap php-xmlrpc httpd php php-mysql php-common php-gd php-odbc
php-pear gcc* net-snmp libssh2 libssh2-devel mariadb-server mariadb mariadb-devel ntpdate
##安装 lamp 和依赖包，使用默认的 centos 7 中的默认 yum 源
[root@server ~]# ntpdate ntp1.aliyun.com
10 Sep 09:08:09 ntpdate[1225]: adjust time server 120.25.115.20 offset -0.000095 sec
[root@server ~]# echo "ntpdate ntp1.aliyun.com" >>/etc/rc.local
[root@server ~]# chmod +x /etc/rc.local
[root@server ~]# systemctl start mariadb
[root@server ~]# systemctl enable mariadb
Created symlink from /etc/systemd/system/multi-user.target.wants/mariadb.service to
/usr/lib/systemd/system/mariadb.service.
[root@server ~]# systemctl start httpd
[root@server ~]# systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to
/usr/lib/systemd/system/httpd.service.
[root@server ~]# mysqladmin -u root password
123123
```

123123

```
[root@server ~]# wget -O zabbix-3.4.2.tar.gz  
http://sourceforge.net/projects/zabbix/files/ZABBIX%20Latest%20Stable/3.4.2/zabbix-3.4.2.tar.gz  
/download    ##下载软件包，上传也可
```

```
[root@server ~]# tar zxvf zabbix-3.4.2.tar.gz
```

```
[root@server ~]# cd zabbix-3.4.2
```

```
[root@server ~]# ./configure --prefix=/usr/local/zabbix --enable-server --enable-agent  
--with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --with-libxml2
```

```
[root@server ~]# make &&make install
```

```
[root@server ~]# mysql -uroot -p123123    ##登陆数据库创建保存监控模板的数据库，并  
导入数据模板，以及授权 zabbix 用户连接数据库；
```

```
create database if not exists zabbix default character set utf8 collate utf8_general_ci;
```

```
use zabbix;
```

```
source /root/zabbix-3.4.2/database/mysql/schema.sql;
```

```
source /root/zabbix-3.4.2/database/mysql/images.sql;
```

```
source /root/zabbix-3.4.2/database/mysql/data.sql;
```

```
grant all on zabbix.* to zabbix@'192.168.100.101' identified by 'zabbix';
```

```
flush privileges;
```

```
show grants for zabbix@'192.168.100.101';
```

```
exit
```

```
[root@server ~]# groupadd zabbix
```

```
[root@server ~]# useradd -r -g zabbix zabbix
```

```
[root@server ~]# mkdir /usr/local/zabbix/logs
```

```
[root@server ~]# chown zabbix:zabbix /usr/local/zabbix/ -R
```

```
[root@server ~]# vi /usr/local/zabbix/etc/zabbix_server.conf
```

```
:%g/^#/d
```

```
:%g/^$/d
```

```
LogFile=/usr/local/zabbix/logs/zabbix_server.log
```

```
PidFile=/tmp/zabbix_server.pid
```

```
DBHost=192.168.100.101
```

```
DBName=zabbix
```

```
DBUser=zabbix
```

```
DBPassword=zabbix
```

```
DBSocket=/var/lib/mysql/mysql.sock
```

```
Include=/usr/local/zabbix/etc/zabbix_server.conf.d/*.conf
```

```
:wq
```

```
[root@server ~]# vi /usr/local/zabbix/etc/zabbix_agentd.conf
```

```
:%g/^#/d
```

```
:%g/^$/d
```

```
PidFile=/tmp/zabbix_agentd.pid
```

```
Server=127.0.0.1,192.168.100.101
```

```
ServerActive=192.168.100.101
```

```
Hostname=server.zabbix.com
```

```
LogFile=/usr/local/zabbix/logs/zabbix_agentd.log
```

```
Include=/usr/local/zabbix/etc/zabbix_agentd.conf.d/*.conf
```

:wq

解释:

PidFile=/tmp/zabbix_agentd.pid	##运行的 PID 文件
Server=127.0.0.1,192.168.100.101	##输入 zabbix_server 主机的 ip 地址
ServerActive=192.168.100.101	##输入 agent 主机采集到数据发送到的 server 端
Hostname=server.zabbix.com	##输入本机的主机名
LogFile=/usr/local/zabbix/logs/zabbix_agentd.log	##指定日志文件
Include=/usr/local/zabbix/etc/zabbix_agentd.conf.d/*.conf	##开启附加配置文件目录

```
[root@server ~]# /usr/local/zabbix/sbin/zabbix_server
```

```
[root@server ~]# netstat -utpln |grep 10051
```

```
tcp        0      0 0.0.0.0:10051          0.0.0.0:*             LISTEN
29079/zabbix_server
```

```
[root@server ~]# /usr/local/zabbix/sbin/zabbix_agentd
```

```
[root@server ~]# netstat -utpln |grep 10050
```

```
tcp        0      0 0.0.0.0:10050          0.0.0.0:*             LISTEN
29148/zabbix_agentd
```

```
[root@server ~]# mv /root/zabbix-3.4.2/frontends/php/ /var/www/html/zabbix
```

```
[root@server ~]# chown -R apache:apache /var/www/html/zabbix/
```

```
[root@server ~]# vi /etc/php.ini
```

```
878 date.timezone = Asia/Shanghai
```

```
384 max_execution_time = 300
```

```
394 max_input_time = 300
```

```
672 post_max_size = 32M
```

```
405 memory_limit = 128M
```

```
854 extension=bcmath.so
```

:wq

解释:

878 date.timezone = Asia/Shanghai	##时区
384 max_execution_time = 300	##单位秒, 通过 POST、GET 以及 PUT 方式接收数据时间进行限制
394 max_input_time = 300	##php 程序上传文件解析数据时的时间限制
672 post_max_size = 32M	##POST 方式请求数据所允许的最大大小
405 memory_limit = 128M	##设置 php 程序的内存限制
854 extension=bcmath.so	##加载 zabbix 程序需要加载的扩展模块 bcmath.so

```
[root@server ~]# systemctl restart httpd
```

2.配置 zabbix 服务的 web 页面并做简单优化;

访问安装 <http://192.168.100.101/zabbix/>, 默认用户名 admin 密码 zabbix

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Welcome to

Zabbix 3.4

Back

Next step

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Check of pre-requisites

	Current value	Required	
PHP version	5.4.16	5.4.0	OK
PHP option "memory_limit"	128M	128M	OK
PHP option "post_max_size"	32M	16M	OK
PHP option "upload_max_filesize"	2M	2M	OK
PHP option "max_execution_time"	300	300	OK
PHP option "max_input_time"	300	300	OK
PHP option "date.timezone"	Asia/Shanghai		OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK
PHP mbstring	on		OK
PHP option "mbstring.func_overload"	off	off	OK

Back

Next step

ZABBIX

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type	<input type="text" value="MySQL"/>
Database host	<input type="text" value="192.168.100.101"/>
Database port	<input type="text" value="3306"/> 0 - use default port
Database name	<input type="text" value="zabbix"/>
User	<input type="text" value="zabbix"/>
Password	<input type="password" value="*****"/>

[Back](#)[Next step](#)

ZABBIX

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Zabbix server details

Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional).

Host	<input type="text" value="192.168.100.101"/>
Port	<input type="text" value="10051"/>
Name	<input type="text"/>

[Back](#)[Next step](#)



- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install**

Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Database type	MySQL
Database server	192.168.100.101
Database port	3306
Database name	zabbix
Database user	zabbix
Database password	*****

Zabbix server	192.168.100.101
Zabbix server port	10051
Zabbix server name	

[Back](#)[Next step](#)

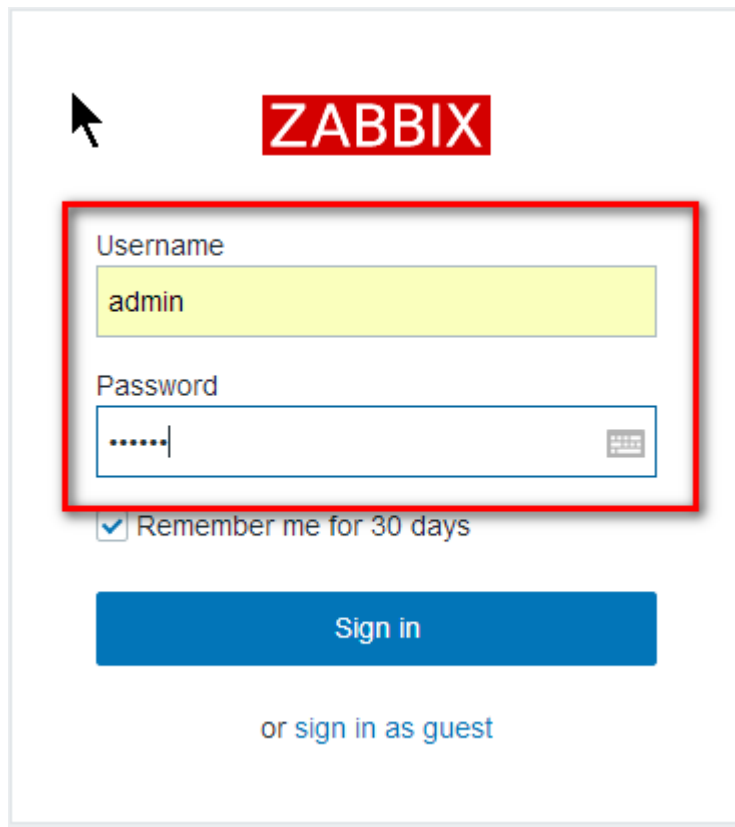
- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install**

Install

Congratulations! You have successfully installed Zabbix frontend.

Configuration file "/var/www/html/zabbix/conf/zabbix.conf.php" created.

[Back](#)[Finish](#)



The image shows the ZABBIX login interface. At the top, there is a ZABBIX logo in a red box. Below it, a red rectangular box highlights the login fields. Inside this box, there is a 'Username' field with the text 'admin' and a 'Password' field with masked characters. Below the password field, there is a checkbox labeled 'Remember me for 30 days'. At the bottom of the red box, there is a blue 'Sign in' button. Below the button, there is a link that says 'or sign in as guest'.

Username

admin

Password

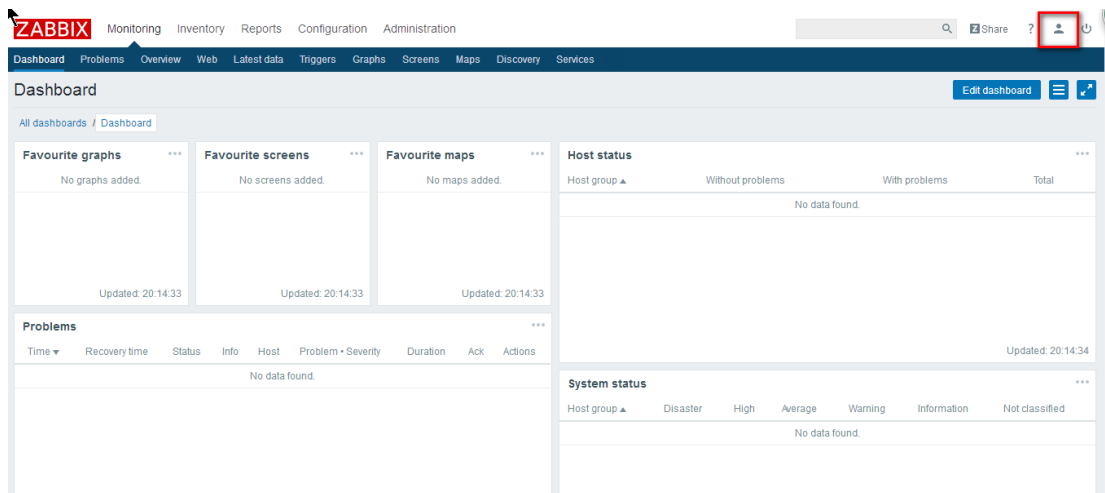
.....

☒ Remember me for 30 days

Sign in

or sign in as guest

安装完成，设置显示字体为中文；



The image shows the ZABBIX dashboard interface. At the top, there is a navigation bar with the ZABBIX logo and various menu items: Monitoring, Inventory, Reports, Configuration, Administration, Dashboard, Problems, Overview, Web, Latest data, Triggers, Graphs, Screens, Maps, Discovery, and Services. Below the navigation bar, there is a 'Dashboard' section with a sub-header 'All dashboards / Dashboard'. The dashboard is divided into several panels. On the left, there are three panels: 'Favourite graphs', 'Favourite screens', and 'Favourite maps', each with a 'No data found.' message and an 'Updated: 20:14:33' timestamp. Below these is a 'Problems' panel with a table header: Time, Recovery time, Status, Info, Host, Problem • Severity, Duration, Ack, Actions. The table body shows 'No data found.' and an 'Updated: 20:14:34' timestamp. On the right, there is a 'Host status' panel with a table header: Host group, Without problems, With problems, Total. The table body shows 'No data found.' and an 'Updated: 20:14:34' timestamp. Below the 'Host status' panel is a 'System status' panel with a table header: Host group, Disaster, High, Average, Warning, Information, Not classified. The table body shows 'No data found.' and an 'Updated: 20:14:34' timestamp.

ZABBIX Monitoring Inventory Reports Configuration Administration

Dashboard Problems Overview Web Latest data Triggers Graphs Screens Maps Discovery Services

Dashboard Edit dashboard

All dashboards / Dashboard

Favourite graphs Favourite screens Favourite maps

No graphs added. No screens added. No maps added.

Updated: 20:14:33 Updated: 20:14:33 Updated: 20:14:33

Problems

Time	Recovery time	Status	Info	Host	Problem • Severity	Duration	Ack	Actions
No data found.								

Updated: 20:14:34

Host status

Host group	Without problems	With problems	Total
No data found.			

Updated: 20:14:34

System status

Host group	Disaster	High	Average	Warning	Information	Not classified
No data found.						

User profile: Zabbix Administrator

User Media Messaging

Password

Language

Theme

Auto-login ☒

Auto-logout ☐ 15m

Refresh

Rows per page

URL (after login)

ZABBIX 监测中 资产记录 报表 配置 管理

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

Dashboard

添加仪表盘 / Dashboard

常用的图形

未添加数据图

已更新: 20:16:31

常用的聚合图形

未添加聚合图形

已更新: 20:16:31

常用的拓扑图

未添加拓扑图

已更新: 20:16:31

主机状态

主机群组	正常设备	异常设备	合计
未发现数据			

已更新: 20:16:32

问题

时间	恢复时间	状态	信息	主机	问题·严重性	持续时间	确认	动作
未发现数据								

系统状态

主机群组	灾难	严重	一般严重	警告	信息	未分类
未发现数据						

解决 zabbix 显示图像下边文字乱码情况:

```
[root@server ~]# cd /var/www/html/zabbix/fonts/
[root@server fonts]#
```

上传字符集文件到此目录下;

```
[root@server fonts]# ls
DejaVuSans.ttf  uming.ttf  字符集文件
```

```
[root@server fonts]# mv DejaVuSans.ttf DejaVuSans.ttf.bak
```

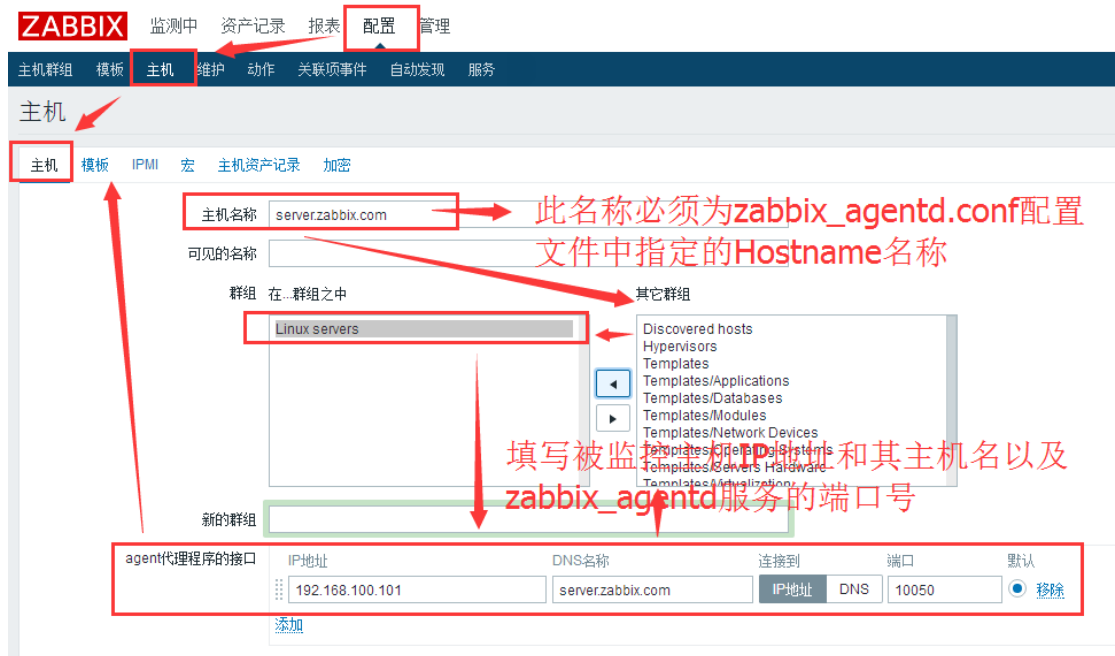
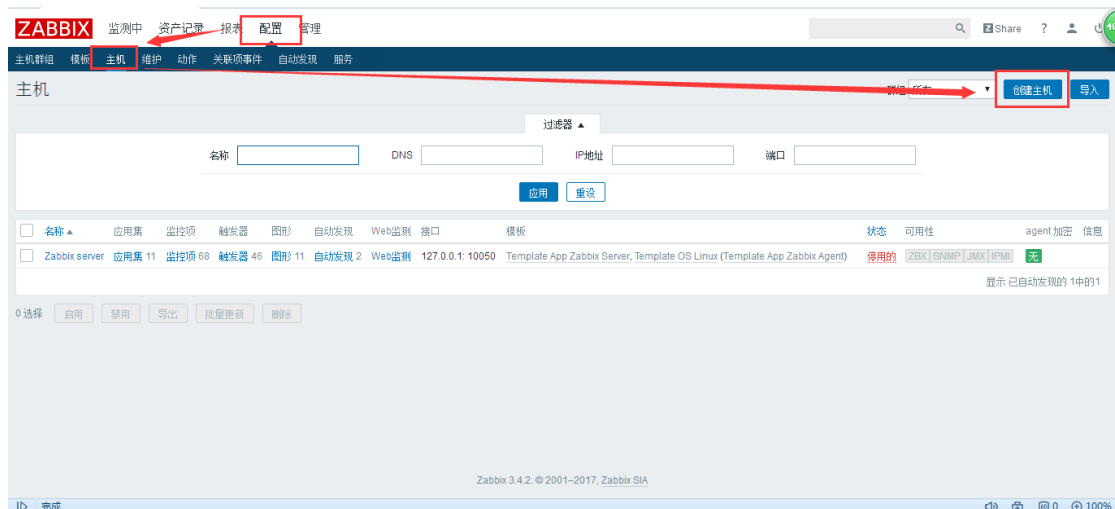
```
[root@server fonts]# mv uming.ttf DejaVuSans.ttf
```

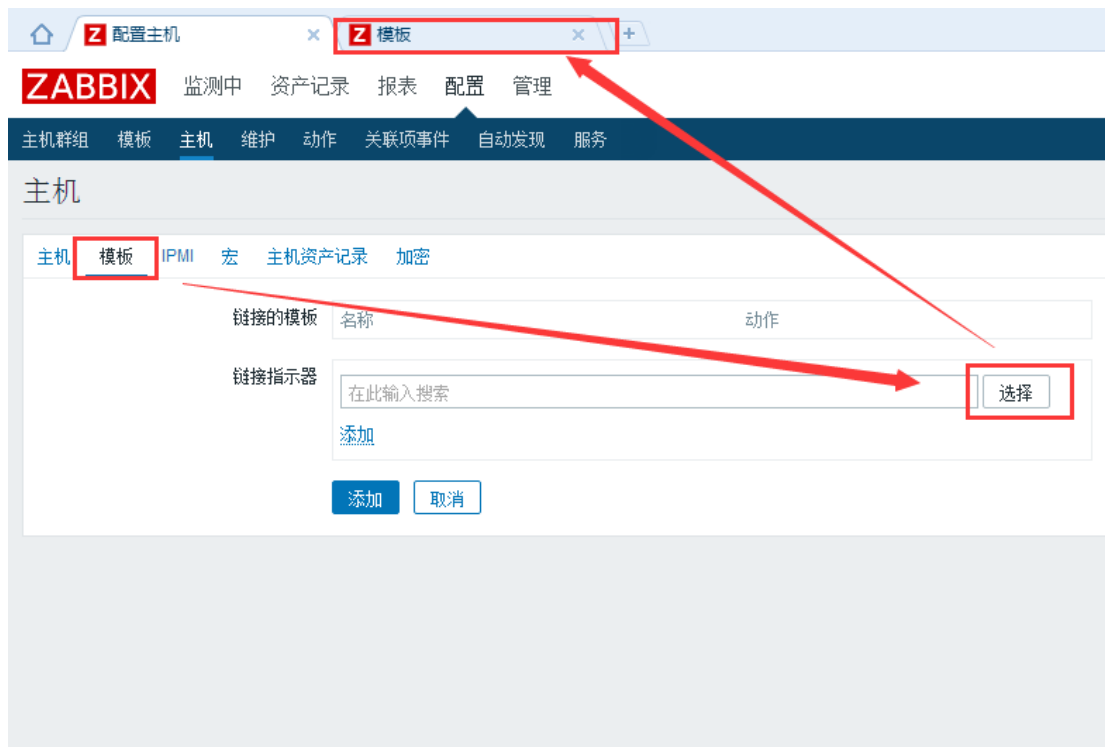
```

[root@server fonts]# chown apache:apache ./*
[root@server fonts]# ll
总用量 36660
-rw-r--r-- 1 apache apache 18387092 11月  5 01:50 DejaVuSans.ttf
-rw-r--r-- 1 apache apache  756072 9月 25 22:17 DejaVuSans.ttf.bak
-rw-r--r-- 1 apache apache 18387092 11月  5 03:00 uming.ttf

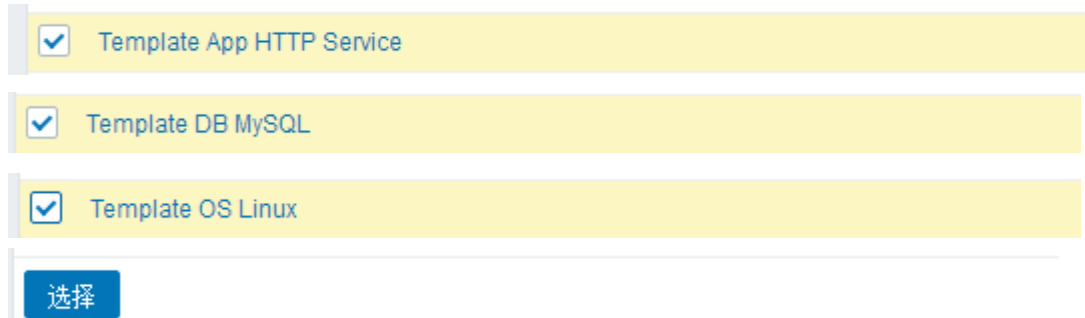
```

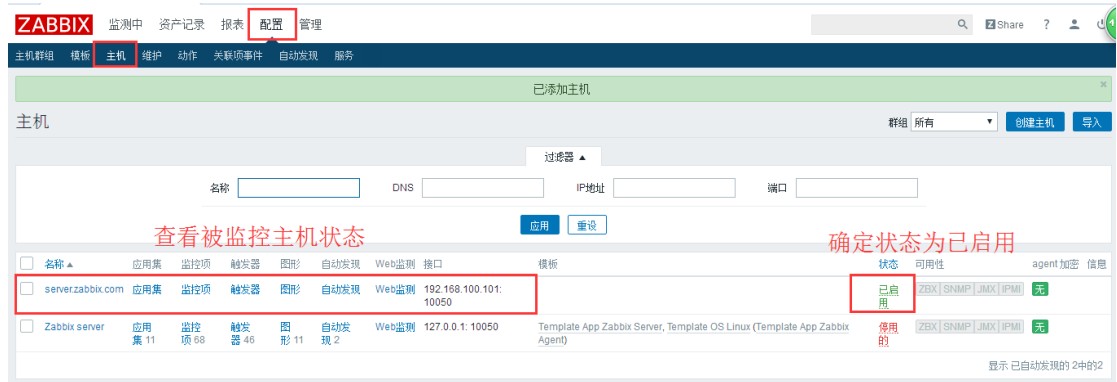
3.实现监控 server.zabbix.com 端的 OS 以及 mysql、httpd;



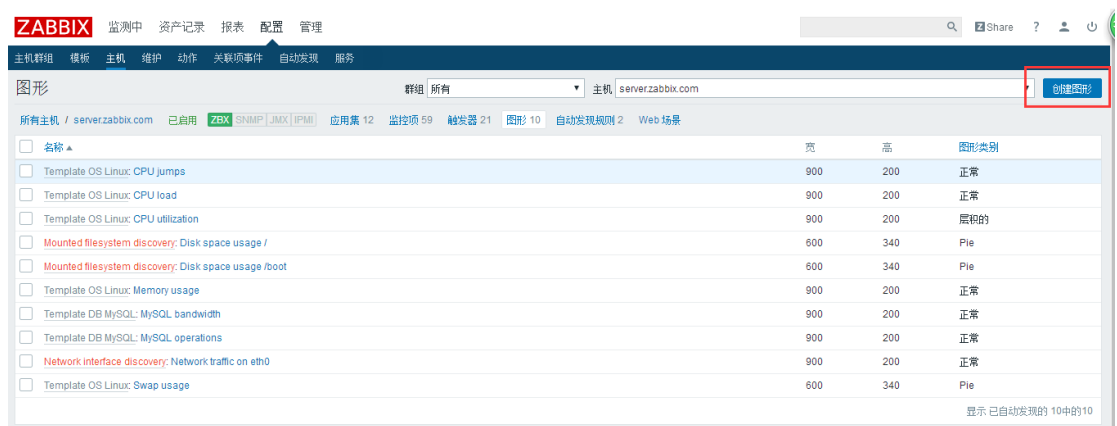


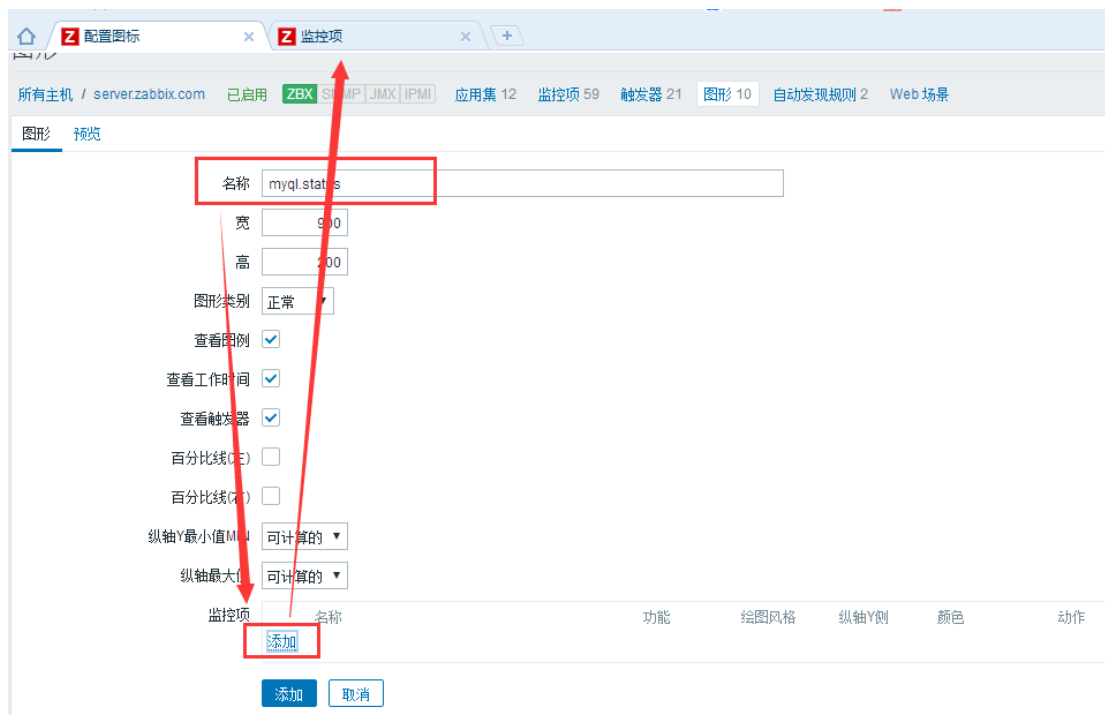
勾选以下三个监控模板，点击选择按钮；



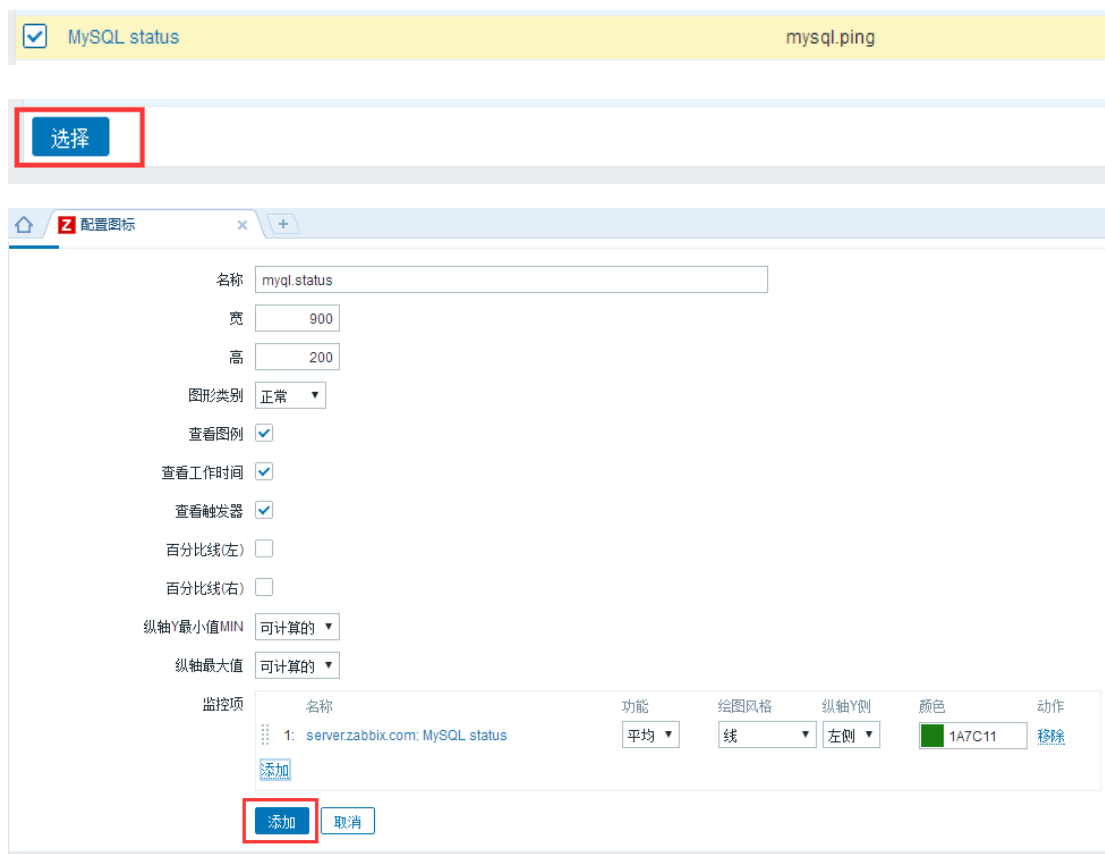


在上个界面会显示此主机的监控项，此主机的图形，可以自行跟据监控项进行创建图形；

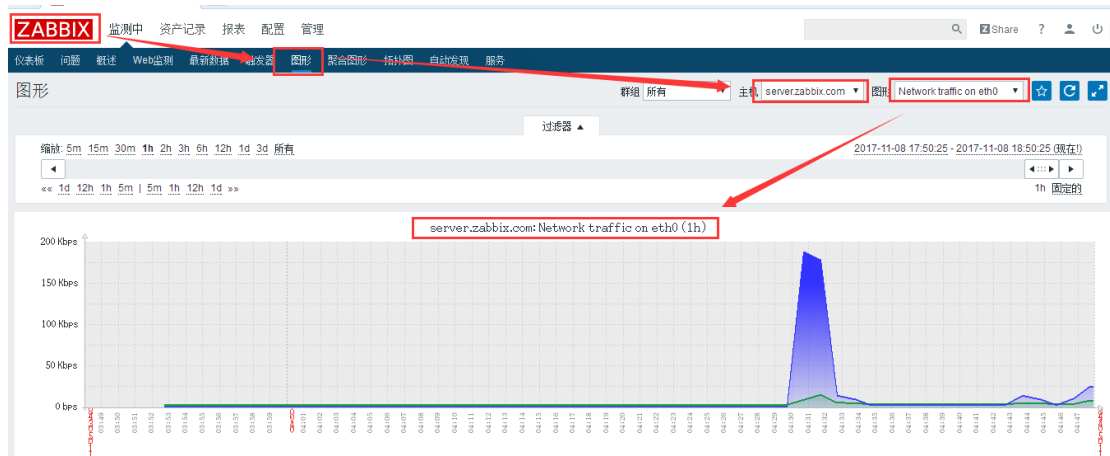




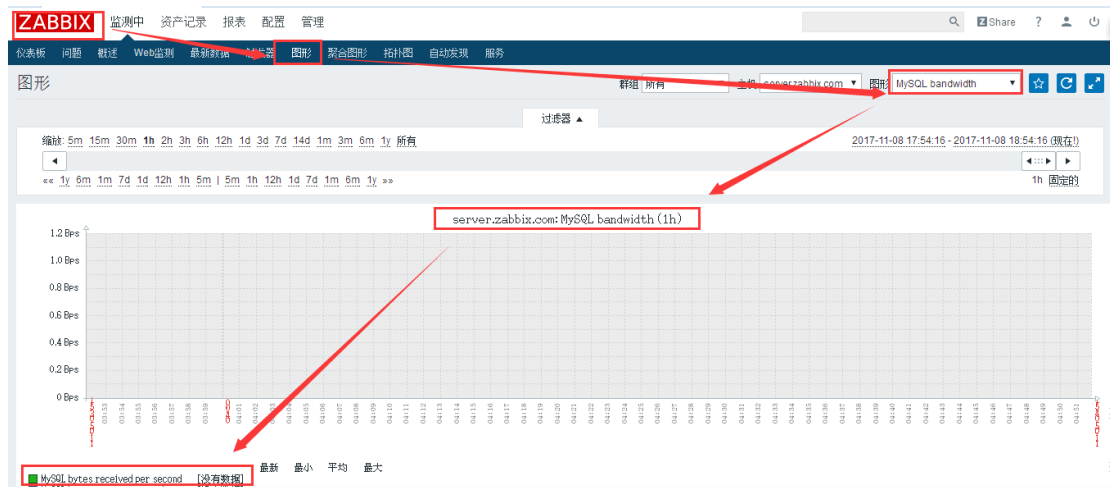
选择监控项，点击选择；



查看监控主机的图形；



但是意外出现了，关于 Mysql 数据库的所有监控项没有图形，如图所示；



经过查看，问题如下图，监控 Mysql 服务器的键值不使用；

The screenshot shows the Zabbix web interface configuration page for a host. The top navigation bar includes 'ZABBIX', '监测中', '资产记录', '报表', '配置', and '管理'. Below this is a secondary navigation bar with '主机群组', '模板', '主机', '维护', '动作', '关联项事件', '自动发现', and '服务'. The main content area is titled '主机' (Hosts). It shows a form for configuring a host. The '名称' (Name) field is empty. The 'DNS' field is empty. The 'IP地址' (IP address) field is empty. The '端口' (Port) field is empty. There are '应用' (Apply) and '重设' (Reset) buttons. Below the form is a table with columns: '名称', '应用集', '监控项', '触发器', '图形', '自动发现', 'Web监测', '接口', and '模板'. The table has two rows. The first row is for 'server.zabbix.com' and the second row is for 'Zabbix server'. The '监控项' (Item) column for 'server.zabbix.com' has a value of '59' and is highlighted with a red box. The '触发器' (Trigger) column for 'server.zabbix.com' has a value of '21' and is highlighted with a red box. The '图形' (Graph) column for 'server.zabbix.com' has a value of '11' and is highlighted with a red box. The '自动发现' (Auto discovery) column for 'server.zabbix.com' has a value of '2' and is highlighted with a red box. The 'Web监测' (Web monitoring) column for 'server.zabbix.com' has a value of '192.168.100.101:10050' and is highlighted with a red box. The '接口' (Interface) column for 'server.zabbix.com' has a value of '10050' and is highlighted with a red box. The '模板' (Template) column for 'server.zabbix.com' has a value of 'Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)' and is highlighted with a red box. The '名称' (Name) column for 'Zabbix server' has a value of '11' and is highlighted with a red box. The '应用集' (Application set) column for 'Zabbix server' has a value of '11' and is highlighted with a red box. The '监控项' (Item) column for 'Zabbix server' has a value of '68' and is highlighted with a red box. The '触发器' (Trigger) column for 'Zabbix server' has a value of '46' and is highlighted with a red box. The '图形' (Graph) column for 'Zabbix server' has a value of '11' and is highlighted with a red box. The '自动发现' (Auto discovery) column for 'Zabbix server' has a value of '2' and is highlighted with a red box. The 'Web监测' (Web monitoring) column for 'Zabbix server' has a value of '127.0.0.1:10050' and is highlighted with a red box. The '接口' (Interface) column for 'Zabbix server' has a value of '10050' and is highlighted with a red box. The '模板' (Template) column for 'Zabbix server' has a value of 'Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)' and is highlighted with a red box. At the bottom of the table, there are buttons for '0 选择', '启用', '禁用', '导出', '批量更新', and '删除'.

名称	应用集	监控项	触发器	图形	自动发现	Web监测	接口	模板
server.zabbix.com	应用集 12	监控项 59	触发器 21	图形 11	自动发现 2	Web监测 192.168.100.101:10050	10050	Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)
Zabbix server	应用集 11	监控项 68	触发器 46	图形 11	自动发现 2	Web监测 127.0.0.1:10050	10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)

<input type="checkbox"/>	***	Template DB MySQL: MySQL begin operations per second	mysql.status[Com_begin]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL bytes received per second	mysql.status[Bytes_received]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL bytes sent per second	mysql.status[Bytes_sent]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL commit operations per second	mysql.status[Com_commit]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL delete operations per second	mysql.status[Com_delete]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL insert operations per second	mysql.status[Com_insert]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL queries per second	mysql.status[Questions]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL rollback operations per second	mysql.status[Com_rollback]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL select operations per second	mysql.status[Com_select]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL slow queries	mysql.status[Slow_queries]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL status	触发器: mysql.ping	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL update operations per second	mysql.status[Com_update]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL uptime	mysql.status[Uptime]	1m	1w	365d	Zabbix 客户端	MySQL	不支持的	
<input type="checkbox"/>	***	Template DB MySQL: MySQL version	mysql.version	1h	1w		Zabbix 客户端	MySQL	不支持的	

解决方案如下；

设置监控失效的键值，打开 **zabbix_server** 服务器，同时在此也是被监控端；

```
[root@server ~]# vi /usr/local/zabbix/etc/zabbix_agentd.conf
```

```
PidFile=/tmp/zabbix_agentd.pid
```

```
Server=127.0.0.1,192.168.100.101
```

```
ServerActive=192.168.100.101
```

```
Hostname=server.zabbix.com
```

```
LogFile=/usr/local/zabbix/logs/zabbix_agentd.log
```

```
Include=/usr/local/zabbix/etc/zabbix_agentd.conf.d/*.conf
```

```
UnsafeUserParameters=1
```

```
UserParameter=mysql.version,mysql -V
```

```
UserParameter=mysql.status[*],/usr/local/zabbix/etc/chk_mysql.sh $1
```

```
UserParameter=mysql.ping,mysqladmin -uroot -p123123 -P3306 -h192.168.100.101 ping | grep
```

```
-c alive
```

```
:wq
```

内容解释：

Server=192.168.100.151,127.0.0.1 //指定 **zabbix** 服务器的地址

ServerActive=192.168.100.101 //指定 **agent** 采集到数据之后发送给 **server** 端

Hostname=server.zabbix.com //指定 **agent** 端的主机名、**server** 端必须需要解析，注意：在 **web** 界面添加被监控主机时，必须写此主机名；

UnsafeUserParameters=1 //允许所有字符的参数传递给用户定义的参数。

UserParameter=mysql.version,mysql -V //定义键值 **mysql.version**，以及键值的值 **mysql -V**

UserParameter=mysql.status[*],/usr/local/zabbix/etc/chk_mysql.sh \$1 //定义键值 **mysql.status[*]**

UserParameter=mysql.ping,mysqladmin -uroot -p123123 -P3306 -h192.168.100.101 ping | grep -c alive ///定义键值 **mysql.ping**，指定 **chk_mysql.sh** 脚本，使用此脚本检查 **mysql** 的运行状态，使用 **mysqladmin** 命令指定 **agent** 端的数据库连接用户密码 **ip** 地址，注意保证 **mysqladmin** 命令的链接；

```
[root@server ~]# vi /usr/local/zabbix/etc/chk_mysql.sh ##编写检查 mysql 状态的脚本，
```

```
在脚本中指定 agent 端的 mysql 的 ip 地址，连接的用户名密码等
```

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

```
# -----
```

```

# FileName:    check_mysql.sh
# Revision:    1.0
# Date:        2015/06/09
# Author:      DengYun
# Email:       dengyun@ttlsa.com
# Website:     www.ttlsa.com
# Description:
# Notes:       ~
# -----
# Copyright:   2015 (c) DengYun
# License:     GPL

# 用户名
MYSQL_USER='root'

# 密码
MYSQL_PWD='123123'

# 主机地址/IP
MYSQL_HOST='192.168.100.101'

# 端口
MYSQL_PORT='3306'

# 数据连接
MYSQL_CONN="/usr/bin/mysqladmin -u${MYSQL_USER} -p${MYSQL_PWD} -h${MYSQL_HOST}
-P${MYSQL_PORT}"

# 参数是否正确
if [ $# -ne "1" ];then
    echo "arg error!"
fi

# 获取数据
case $1 in
    Uptime)
        result=`${MYSQL_CONN} status|cut -f2 -d":"|cut -f1 -d"T"`
        echo $result
        ;;
    Com_update)
        result=`${MYSQL_CONN} extended-status |grep -w "Com_update"|cut -d"|" -f3`
        echo $result
        ;;
    Slow_queries)

```

```

        result=`${MYSQL_CONN} status | cut -f5 -d":" | cut -f1 -d"O"`
        echo $result
        ;;
Com_select)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_select" | cut -d"|" -f3`
        echo $result
        ;;
Com_rollback)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_rollback" | cut -d"|" -f3`
        echo $result
        ;;
Questions)
        result=`${MYSQL_CONN} status | cut -f4 -d":" | cut -f1 -d"S"`
        echo $result
        ;;
Com_insert)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_insert" | cut -d"|" -f3`
        echo $result
        ;;
Com_delete)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_delete" | cut -d"|" -f3`
        echo $result
        ;;
Com_commit)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_commit" | cut -d"|" -f3`
        echo $result
        ;;
Bytes_sent)
        result=`${MYSQL_CONN} extended-status | grep -w "Bytes_sent" | cut -d"|" -f3`
        echo $result
        ;;
Bytes_received)
        result=`${MYSQL_CONN} extended-status | grep -w "Bytes_received" | cut -d"|" -f3`
        echo $result
        ;;
Com_begin)
        result=`${MYSQL_CONN} extended-status | grep -w "Com_begin" | cut -d"|" -f3`
        echo $result
        ;;

*)
        echo
        "Usage:$0(Uptime|Com_update|Slow_queries|Com_select|Com_rollback|Questions|Com_inser
t|Com_delete|Com_commit|Bytes_sent|Bytes_received|Com_begin)"

```

```

;;
esac
:wq
[root@server ~]# chmod 777 /usr/local/zabbix/etc/chk_mysql.sh
[root@server ~]# mysql -uroot -p123123
grant all on *.* to 'root'@'192.168.100.101' identified by '123123';
flush privileges;
show grants for root@'192.168.100.101';
exit
[root@server ~]# ln -s /usr/local/zabbix/bin/* /usr/local/bin/      //为 zabbix 的命令做软链
接，检测 mysql 状态时会使用 zabbix_get 命令
[root@server ~]# yum -y install psmisc
[root@server ~]# killall -9 zabbix_server
[root@server ~]# killall -9 zabbix_agentd
[root@server ~]# /usr/local/zabbix/sbin/zabbix_server
[root@server ~]# /usr/local/zabbix/sbin/zabbix_agentd
[root@server ~]# netstat -utpln |grep 100
tcp        0      0 0.0.0.0:10050          0.0.0.0:*              LISTEN
29800/zabbix_agentd
tcp        0      0 0.0.0.0:10051          0.0.0.0:*              LISTEN
29733/zabbix_server

[root@server ~]# vi /etc/my.cnf      ##禁止反向解析
[mysqld]
skip-name-resolve
[root@server ~]# systemctl restart mariadb
[root@server ~]# zabbix_get -s 192.168.100.101 -k mysql.ping      ##在此可以通过 zabbix_get
命令进行获取服务器监控项的具体键值，mysql.ping 是在 zabbix_agentd.conf 配置文件中的键
值名称，通过此名称可以查看键值，通常能够获取到键值，获取监控数据就是没问题的；
1
[root@server ~]# zabbix_get -s 192.168.100.101 -k mysql.status[Com_update]
143
[root@server ~]# zabbix_get -s 192.168.100.101 -k mysql.version
mysql  Ver 15.1 Distrib 5.5.56-MariaDB, for Linux (x86_64) using readline 5.1

```

再次打开 zabbix 的 web 界面，验证 mysql 服务器的监控界面；



按照如上方法将 **server.linuxfan.cn** 的 **httpd** 服务的状态添加到 **zabbix** 的图形中；

4. 安装及配置 **zabbix_agentd** 代理端（FTP 服务器）；

首先安装 **zabbix_agentd** 端的测试服务（**ftp**）并且确保网络通信；

```
[root@agent ~]# ip a
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
```

```
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

```
inet 127.0.0.1/8 scope host lo
```

```
inet6 ::1/128 scope host
```

```
valid_lft forever preferred_lft forever
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
```

```
link/ether 00:0c:29:a0:76:46 brd ff:ff:ff:ff:ff:ff
```

```
inet 192.168.100.152/24 brd 192.168.100.255 scope global eth0
```

```
inet6 fe80::20c:29ff:fea0:7646/64 scope link
```

```
valid_lft forever preferred_lft forever
```

```
[root@agent ~]# vi /etc/sysconfig/network
```

```
NETWORKING=yes
```

```
HOSTNAME=agent.zabbix.com
```

```
:wq
```

```
[root@agent ~]# vi /etc/hosts
```

```
192.168.100.101 server.zabbix.com
```

```
192.168.100.152 agent.zabbix.com
```

```
192.168.100.153 win2k8dc1
```

```
:wq
```

```
[root@agent ~]# reboot
```

```
[root@agent ~]# ping server.zabbix.com -c 4
```

```
[root@agent ~]# yum -y install vsftpd
```

```
[root@agent ~]# /etc/init.d/vsftpd start
```

其次开始配置 **agent.zabbix.com** 被监控端主机的 **zabbix** 的 **agent** 代理端；

```

[root@agent ~]# wget -O /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.aliyun.com/repo/Centos-7.repo
[root@agent ~]# yum -y install libxml2-devel libcurl-devel pcre-devel ntpdate
[root@agent ~]# ntpdate ntp1.aliyun.com
[root@agent ~]# wget -O zabbix-3.4.2.tar.gz
http://sourceforge.net/projects/zabbix/files/ZABBIX%20Latest%20Stable/3.4.2/zabbix-3.4.2.tar.gz
/download
[root@agent ~]# tar zxvf zabbix-3.4.2.tar.gz
[root@agent ~]# cd zabbix-3.4.2
[root@agent zabbix-3.4.2]# ./configure --prefix=/usr/local/zabbix --enable-agent --enable-ipv6
--with-net-snmp --with-libcurl --with-libxml2
[root@agent zabbix-3.4.2]# make &&make install
[root@agent zabbix-3.4.2]# cp misc/init.d/tru64/zabbix_agentd /etc/init.d/
[root@agent zabbix-3.4.2]# vi /etc/init.d/zabbix_agentd
24 DAEMON=/usr/local/zabbix/sbin/zabbix_agentd
:wq
[root@agent zabbix-3.4.2]# chmod +x /etc/init.d/zabbix_agentd
[root@agent zabbix-3.4.2]# cd
[root@agent ~]# groupadd zabbix
[root@agent ~]# useradd -r -g zabbix zabbix
[root@agent ~]# chown -R zabbix:zabbix /usr/local/zabbix/
[root@agent ~]# vi /usr/local/zabbix/etc/zabbix_agentd.conf
:%g/^$/d
:%g/^#/d
LogFile=/tmp/zabbix_agentd.log
Server=192.168.100.101
ServerActive=192.168.100.101
Hostname=agent.zabbix.com
:wq
[root@agent ~]# /etc/init.d/zabbix_agentd start
Zabbix agent started.
[root@agent ~]# netstat -utpln |grep zabbix
tcp        0      0 0.0.0.0:10050          0.0.0.0:*             LISTEN
12481/zabbix_agentd
tcp        0      0 :::10050              :::*                   LISTEN
12481/zabbix_agentd

```

5.登录 zabbix 的 web 界面，进行添加 agent 主机，验证监控效果；

The image shows the Zabbix web interface for adding a new host. The process is as follows:

- Step 1: Hosts Page**
 - Navigation: **配置** (Configuration) > **主机** (Hosts).
 - Action: Click **创建主机** (Create host).
- Step 2: Host Configuration Form**
 - Fields to fill: **主机名称** (Host name) to `agentzabbix.com`, **可见的名称** (Visible name) to `agentzabbix.com`.
 - Group selection: Select **Linux servers** from the **在...群组之中** (In... group) list.
 - Agent interface: **agent代理程序的接口** (Agent interface) to `192.168.100.152`, **DNS名称** (DNS name) to `agentzabbix.com`, **连接到** (Connect to) to `IP地址` (IP address), **端口** (Port) to `10050`.
 - Action: Click **添加** (Add).
- Step 3: Template Selection**
 - Navigation: **配置** (Configuration) > **模板** (Templates).
 - Selection: Check **Template App FTP Service** and **Template OS Linux**.
 - Action: Click **选择** (Select).



<input type="checkbox"/>	名称	应用集	监控项	触发器	图形	自动发现	Web监测	接口	模板	状态	可用性	agent加密	信息
<input type="checkbox"/>	agent.zabbix.com	应用集 11	监控项 33	触发器 16	图形 5	自动发现 2	Web监测	192.168.100.152:10050	Template App FTP Service, Template OS Linux (Template App Zabbix Agent)	已启用	ZBX SNMP JMX IPMI	无	
<input type="checkbox"/>	server.zabbix.com	应用集 12	监控项 59	触发器 21	图形 11	自动发现 2	Web监测	192.168.100.101:10050	Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)	已启用	ZBX SNMP JMX IPMI	无	
<input type="checkbox"/>	Zabbix server	应用集 11	监控项 68	触发器 46	图形 11	自动发现 2	Web监测	127.0.0.1:10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)	停用的	ZBX SNMP JMX IPMI	无	

显示 已自动发现的 3中的3

添加 agent.zabbix.com 主机完成，默认的图形之后五个，无 ftp 服务图形，接下来手动添加 ftp 服务图形显示；



<input type="checkbox"/>	名称	应用集	监控项	触发器	图形	自动发现	Web监测	接口	模板
<input type="checkbox"/>	agent.zabbix.com	应用集 11	监控项 33	触发器 16	图形 5	自动发现 2	Web监测	192.168.100.152:10050	Template App FTP Service, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	server.zabbix.com	应用集 12	监控项 59	触发器 21	图形 11	自动发现 2	Web监测	192.168.100.101:10050	Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	Zabbix server	应用集 11	监控项 68	触发器 46	图形 11	自动发现 2	Web监测	127.0.0.1:10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)

显示 已自动发现的 3中的3

<input type="checkbox"/>	名称	应用集	监控项	触发器	图形	自动发现	Web监测	接口	模板
<input type="checkbox"/>	agent.zabbix.com	应用集 11	监控项 33	触发器 16	图形 5	自动发现 2	Web监测	192.168.100.152:10050	Template App FTP Service, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	server.zabbix.com	应用集 12	监控项 59	触发器 21	图形 11	自动发现 2	Web监测	192.168.100.101:10050	Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	Zabbix server	应用集 11	监控项 68	触发器 46	图形 11	自动发现 2	Web监测	127.0.0.1:10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)

显示 已自动发现的 3中的3

<input type="checkbox"/>	名称	应用集	监控项	触发器	图形	自动发现	Web监测	接口	模板
<input type="checkbox"/>	agent.zabbix.com	应用集 11	监控项 33	触发器 16	图形 5	自动发现 2	Web监测	192.168.100.152:10050	Template App FTP Service, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	server.zabbix.com	应用集 12	监控项 59	触发器 21	图形 11	自动发现 2	Web监测	192.168.100.101:10050	Template App HTTP Service, Template DB MySQL, Template OS Linux (Template App Zabbix Agent)
<input type="checkbox"/>	Zabbix server	应用集 11	监控项 68	触发器 46	图形 11	自动发现 2	Web监测	127.0.0.1:10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)

显示 已自动发现的 3中的3

配置图标

监控项

所有主机 / agentzabbix.com 已启用 ZBX SNMP JMX IPMI 应用集 11 监控项 33 触发器 16 图形 5 自动发现规则 2 Web 场景

图形 预览

名称 ftp.status

宽 900

高 200

图形类别 正常

查看图例 ☒

查看工作时间 ☒

查看触发器 ☒

百分比线(左) ☐

百分比线(右) ☐

纵轴Y最小值MIN 可计算的

纵轴最大值 可计算的

监控项

名称

功能

绘图风格

纵轴Y侧

颜色

添加

添加 取消

☒ FTP service is running net.tcp.service[ftp]

选择

名称 ftp.status

宽 900

高 200

图形类别 正常

查看图例 ☒

查看工作时间 ☒

查看触发器 ☒

百分比线(左) ☐

百分比线(右) ☐

纵轴Y最小值MIN 可计算的

纵轴最大值 可计算的

监控项

名称

功能

绘图风格

纵轴Y侧

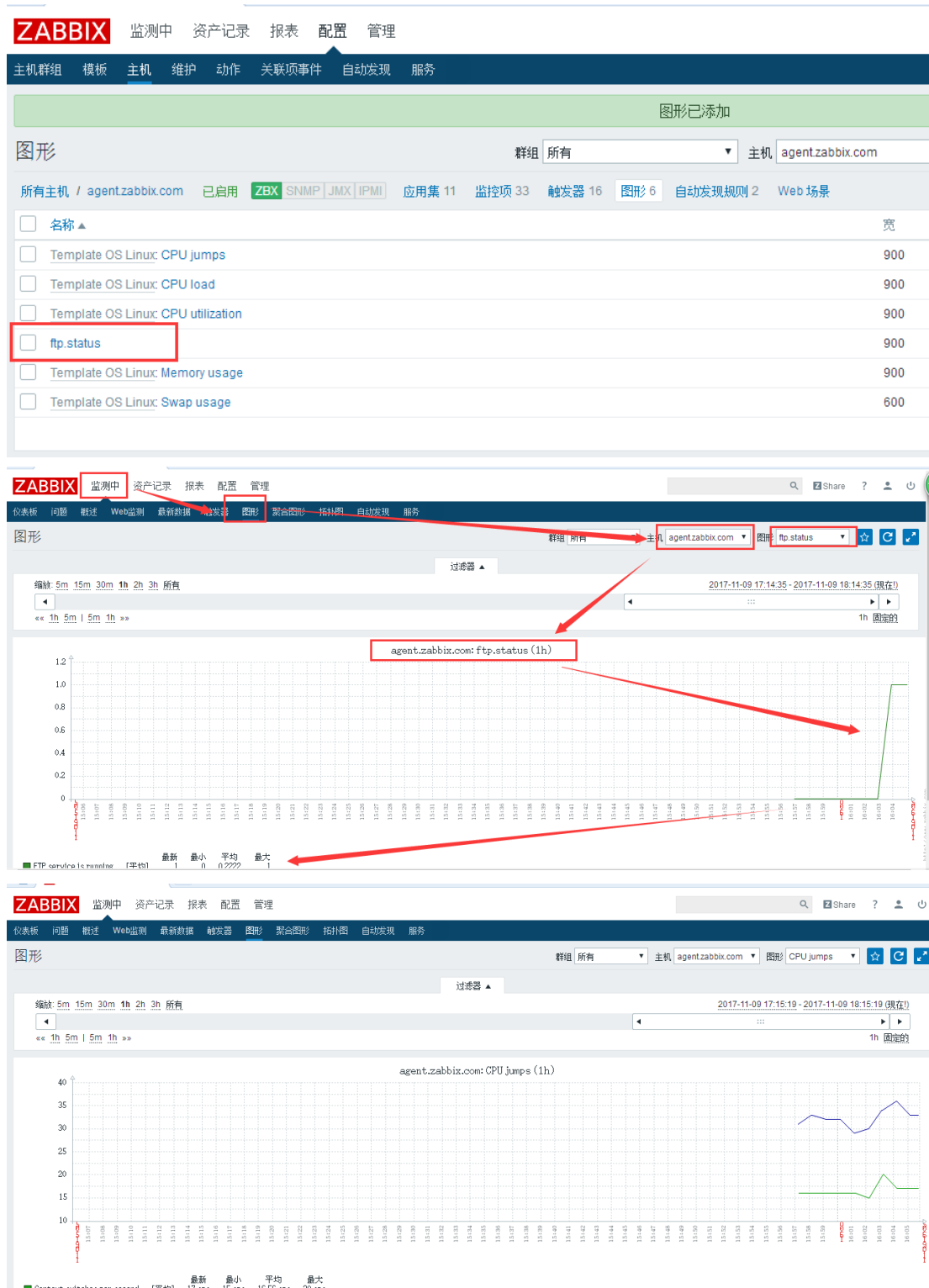
颜色

动作

添加

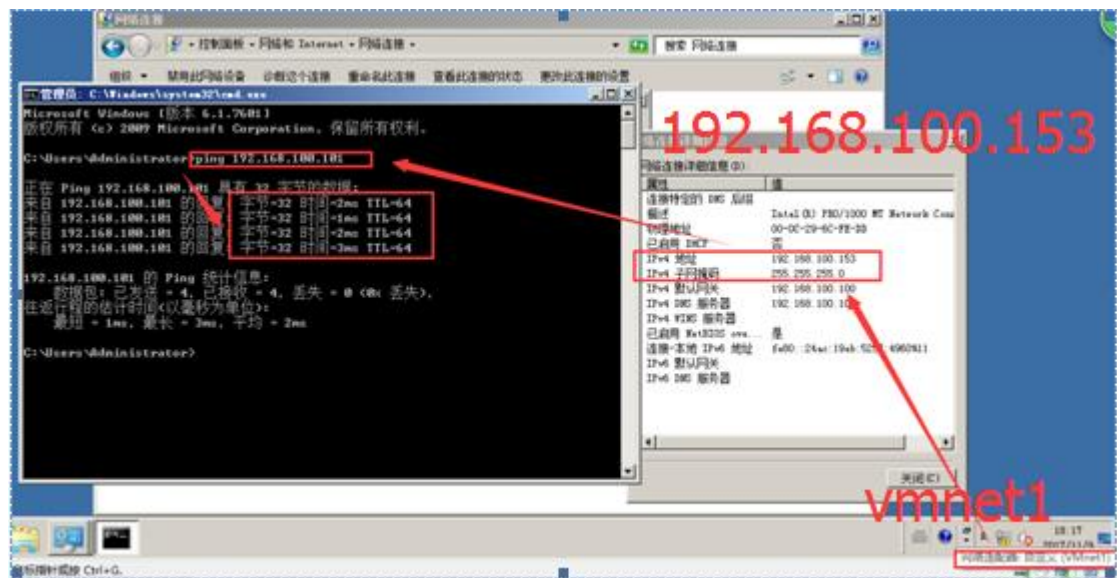
1: agentzabbix.com: FTP service is running 平均 线 左侧 1A7C11 移除

添加 取消



6.配置监控 Win2k8dc1 主机，验证效果；

首先配置 windows2008 主机的网络；



其次用宿主机访问 [zabbix](https://www.zabbix.com/download) 官网下载监控 windows2008 的插件；

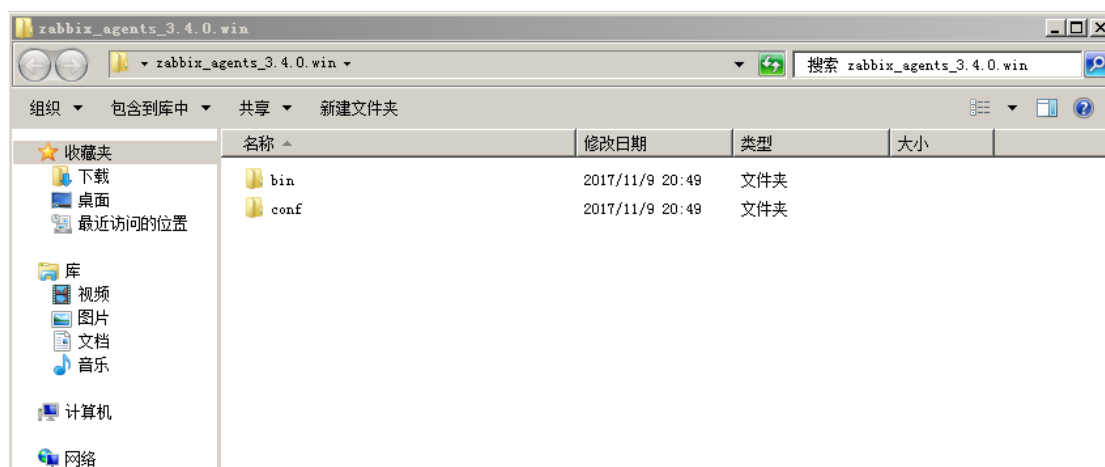


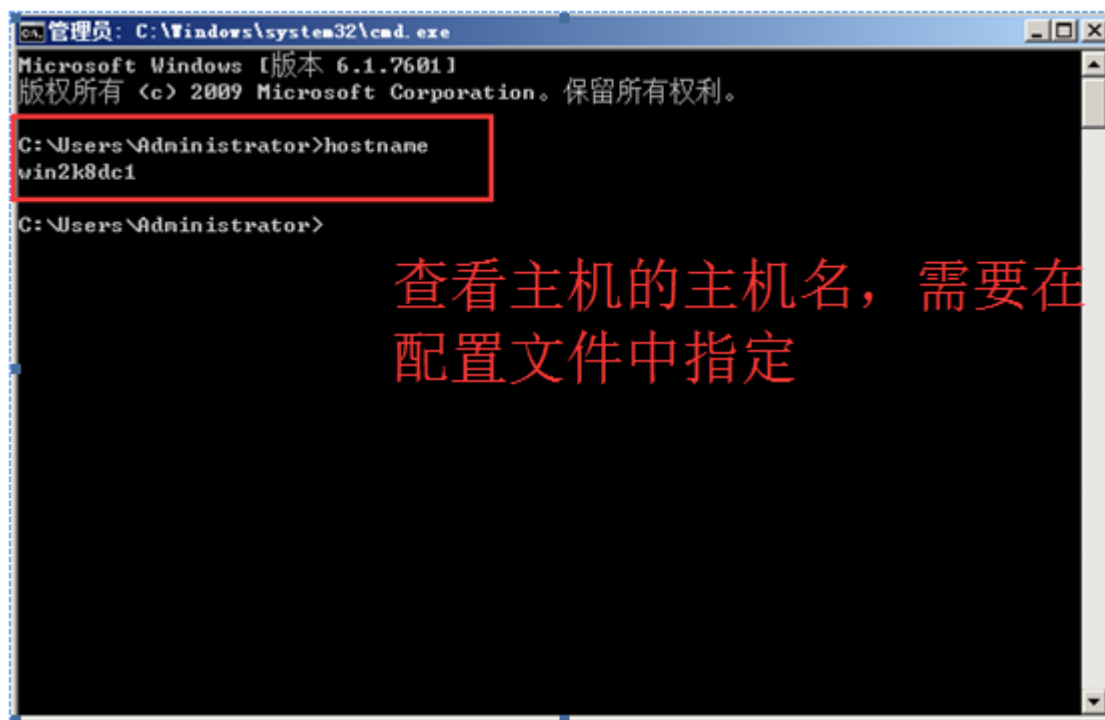
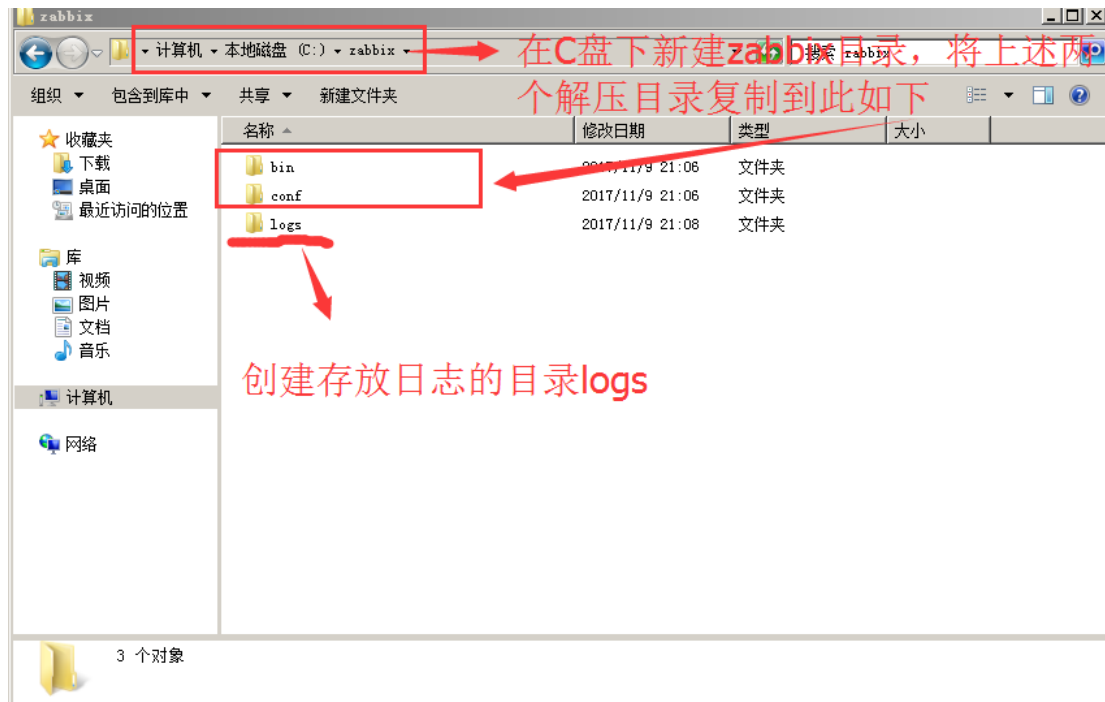
Zabbix pre-compiled agents

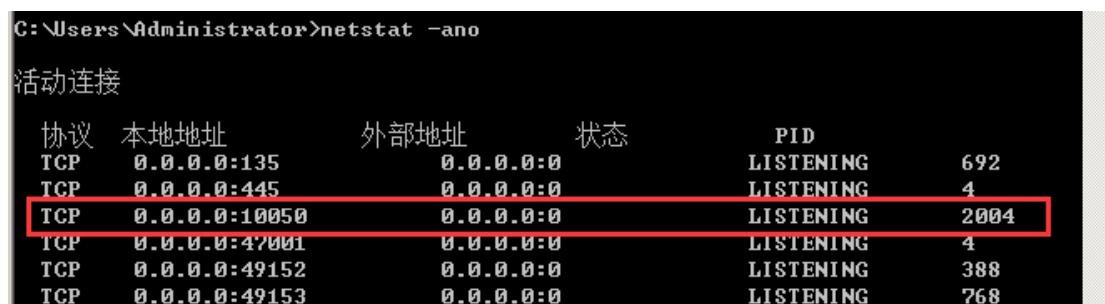
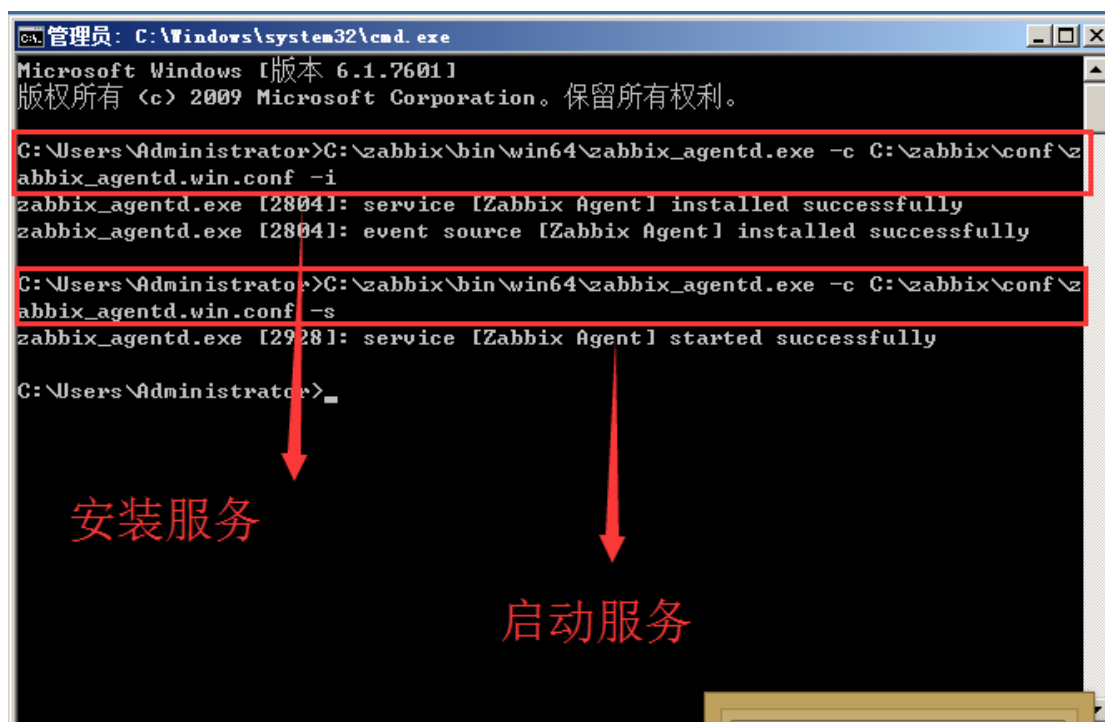
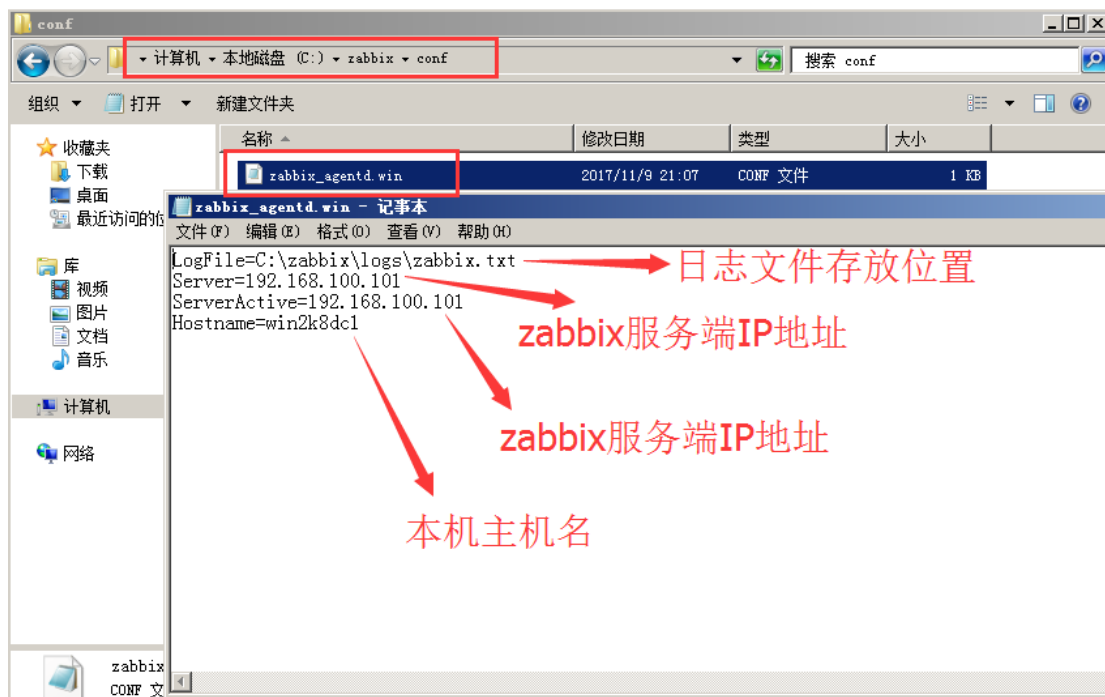
Package	Release	OS	Hardware	MD5	Download
Zabbix 3.4	3.4.0	AIX 7.1	powerpc	5b53fa8861cea6e13a0855f92972ef7d	Download
		FreeBSD 11.1	amd64	6206bd0710805338602b343db58543ac	Download
		FreeBSD 11.1	i386	ac473e92013f7a56adf729bd952fc7e8	Download
		OpenBSD 5.9	amd64	03a48c067aedde117e825ff7af931e6	Download
		OpenBSD 5.9	i386	2f53f6ef6104b5069ed24753137b1144	Download
		OpenBSD 6.1	amd64	198e164e83a6bd836250e6e78c1f5adf	Download
		OpenBSD 6.1	i386	9ce590e5dc717860a09fbd3f3d782307	Download
		Windows (All)	i386,amd64	c844676d004c73deeb8c886ee463593f	Download
		AIX 7.1	powerpc	3143e7c28de039608132f2e3c6fa9ec0	Download
		FreeBSD 11.1	amd64	ac5d558fefa7dfc760b4e74eba1dfe30	Download
	3.2.7	FreeBSD 11.1	i386	1aab24494811b43fbd7a957a56f06dd	Download
		Linux 2.4.x	i386	af4956820460a2da81c95d1cefc0779e	Download
		Linux 2.6	amd64	9e06786d23f0ed33f6fca963379ab52	Download
		Linux 2.6	i386	35712d3363cd24feb597e60ece7b5658	Download
		OpenBSD 5.9	amd64	0e770875d2a8ec6e53bb12df44f5a36	Download
		OpenBSD 5.9	i386	5e243caba729727bb566aa8bc002615d	Download
		OpenBSD 6.1	amd64	1893c16ebd0137c0b2c5b19b0476c31e	Download

下载完成，将下载后的插件复制到测试主机 win2k8dc1 中，进行配置；

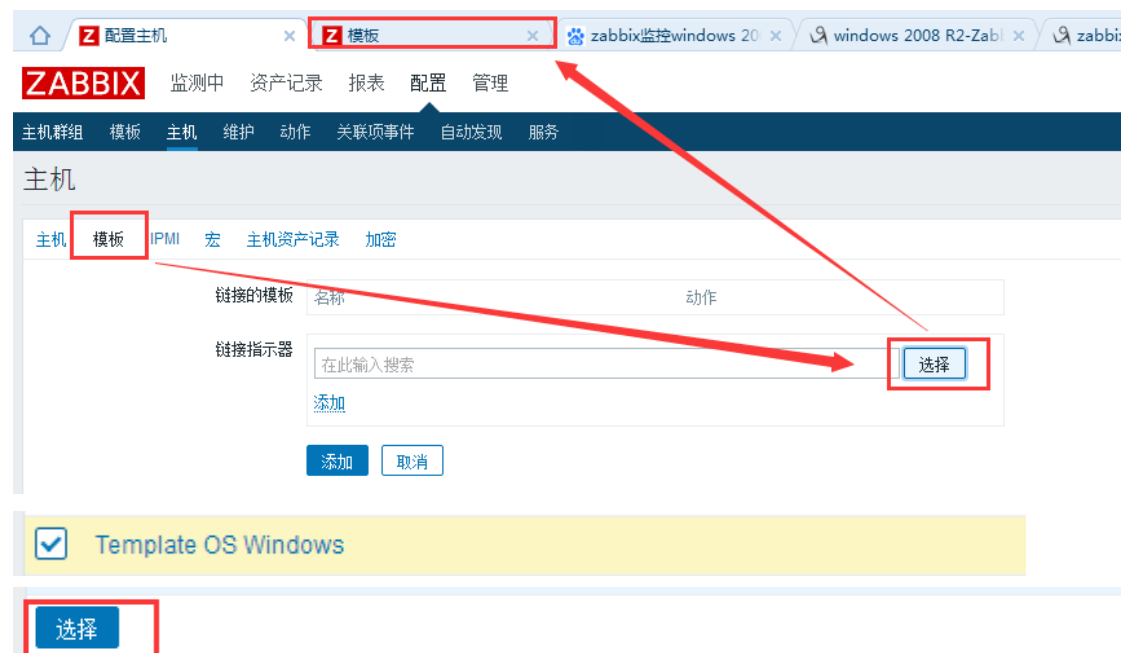
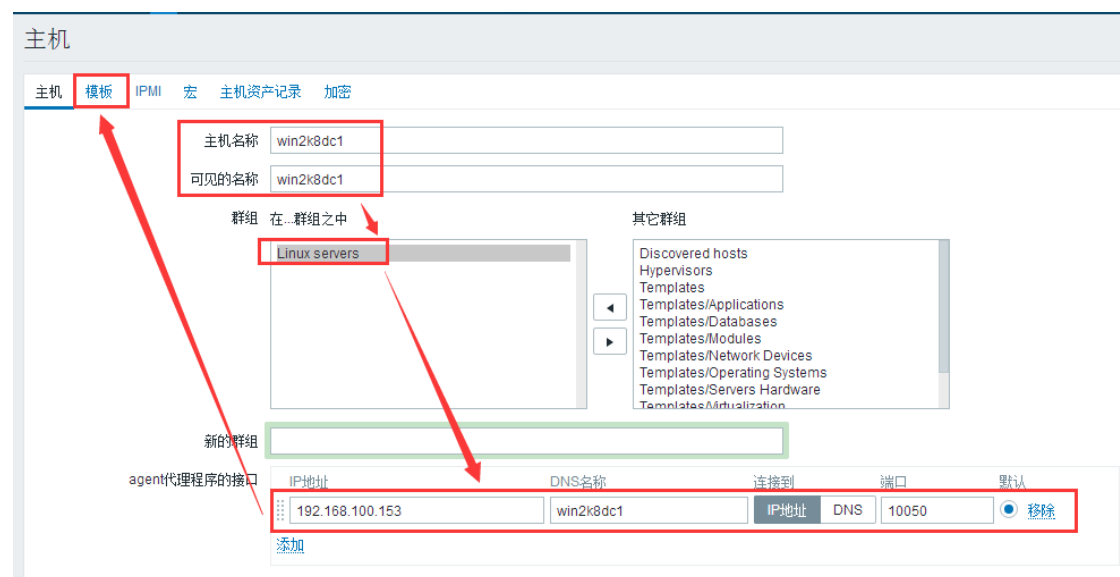
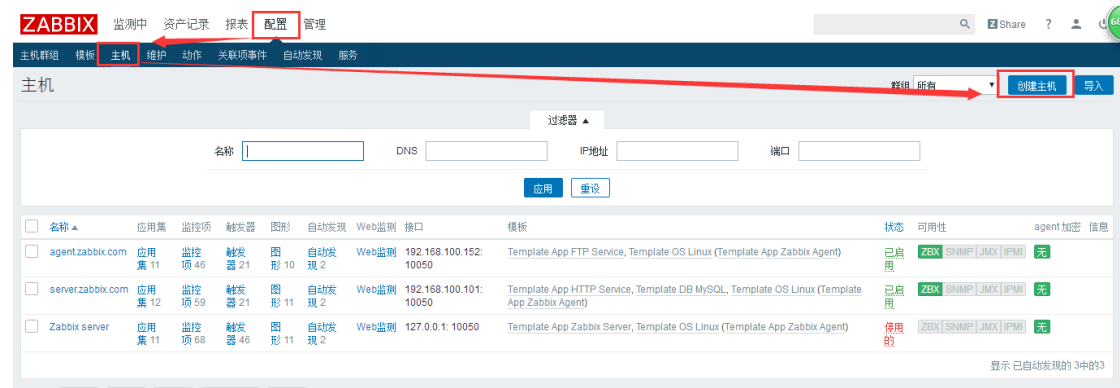


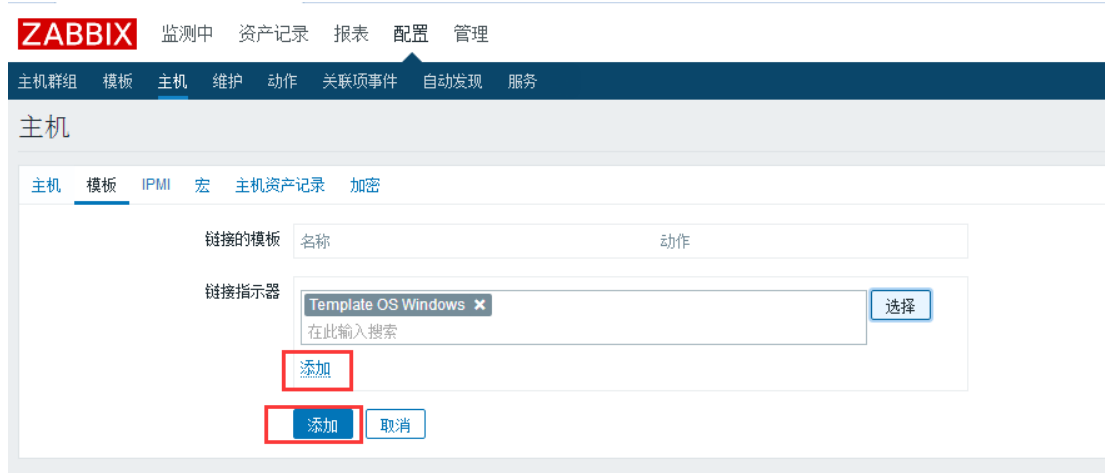




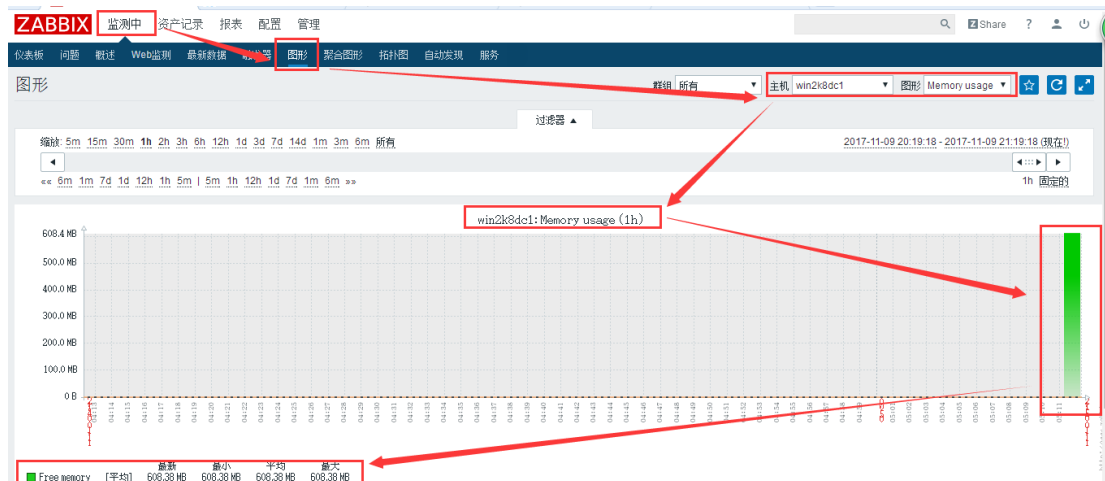


C:\zabbix\bin\win64\zabbix_agentd.exe -c C:\zabbix\conf\zabbix_agentd.win.conf -i ##安装
 C:\zabbix\bin\win64\zabbix_agentd.exe -c C:\zabbix\conf\zabbix_agentd.win.conf -s ##启动
 C:\zabbix\bin\win64\zabbix_agentd.exe -c C:\zabbix\conf\zabbix_agentd.win.conf -d ##删除
windows 主机配置完成，下边在 web 界面，添加主机 win2k8dc1；





添加完成，默认图形为 2 个，可以手动添加，在此不做过多说明，验证当前图形如下：



7.自定义监控项，创建项目、触发器、图形，验证监控效果；

以上关联的 [Template OS Linux](#) 模板基本涵盖了所有系统层面的监控，包括了我们最关注的

几项：ping、load、cpu 使用率、memory、disk、网卡流量等等，当然有些触发器的阈值可能需要根据服务器的自身情况进行修改。

下面添加自定义服务器内存使用检测项，在此配置一个监控项为：内存使用 70M 进行警告提醒：

1.修改客户端 zabbix_agentd.conf 配置文件，最后一行添加：

UserParameter=memory_userd,free -m|grep Mem|awk '{print \$3}'

语法：UserParameter=key,shell command

监控 key 值：memory_userd，key 值可以随意编写，但是一会需要在 web 页面创建监控项时指定 key 值；

Shell 命令或脚本：free -m|grep Mem|awk '{print \$3}'

注：在 zabbix_server 端可以使用 zabbix_get -s agent 端 ip 地址 -p 10050 -k key 名

通过此命令可以查看 agent 端 key 的监控值：

具体实现方式如下，更改 agent.zabbix.com 主机的配置文件来自定义监控项；

```
[root@agent ~]# vi /usr/local/zabbix/etc/zabbix_agentd.conf
```

```
    UserParameter=memory_userd,free -m|grep Mem|awk '{print $3}'
```

```
:wq
```

```
[root@agent ~]# /etc/init.d/zabbix_agentd restart
```

使用 server.agent.com 主机查看其监控项键值；

```
[root@server ~]# zabbix_get -s 192.168.100.152 -k memory_userd
```

338

使用 web 界面添加此监控项；





监控项

所有主机 / agent.zabbix.com 已启用 ZBX SNMP JMX IPMI 应用集 11 监控项 33 触发器 16 图形 6 自动发现规则 2 Web 场景

监控项 进程

名称 **监控项名称自定义编写**

类型

键值 **键值名称必须为被监控端配置文件中定义的**

主机接口

信息类型

单位

更新间隔

自定义时间间隔

类型	间隔	期间	动作
灵活	调度	50s	1-7,00:00-24:00

[添加](#)

历史数据保留时长

趋势存储时间

查看值 [展示值映射](#)

新的应用集

应用集

填入主机资产纪录栏位

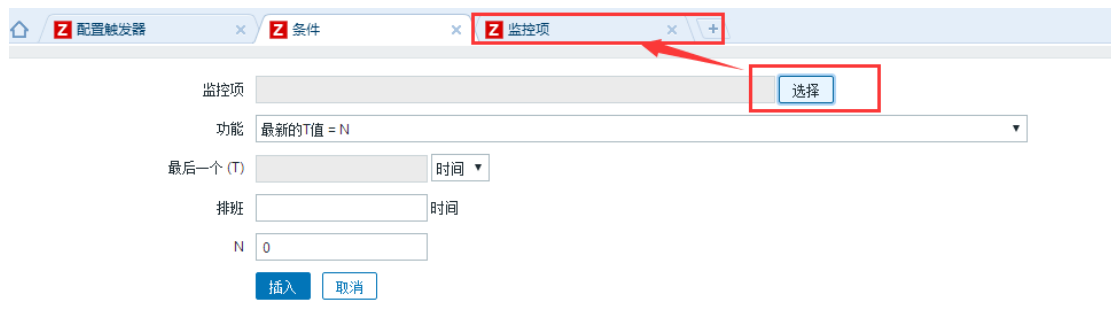
描述

已启用 ☒

添加 **取消**



监控项添加完成后，创建触发器（根据条件进行触发某个告警操作）；



名称	键值
Agent ping	agent.ping
agent_memory_used	memory_userd
Available memory	vm.memory.size[available]
Checksum of /etc/passwd	vfs.file.cksum[/etc/passwd]
Context switches per second	system.cpu.switches
CPU idle time	system.cpu.util[idle]
CPU interrupt time	system.cpu.util[interrupt]
CPU iowait time	system.cpu.util[iowait]
CPU nice time	system.cpu.util[nice]

配置触发器

条件

监控项

agent.zabbix.com: agent_memory_used

选择

功能

最新的T值 = N

最后一个 (T)时间

排班时间

N

0

插入

取消

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 关联项事件 自动发现 服务

触发器

所有主机 / agent.zabbix.com 已启用 ZBX SNMP JMX IPMI 应用集 11 监控项 46 触发器 20 图形 9 自动发现规则 2 Web 场景

触发器 依赖关系

名称

agent_memory>200M

严重性

未分类

信息

警告

一般严重

严重

灾难

表达式

{agent.zabbix.com:memory_userd.last()}>200

添加

表达式构造器

事件成功迭代

表达式

恢复表达式

无

问题事件生成模式

单个

多重

事件成功关闭

所有问题

所有问题如果标签值匹配

标记

标记

值

移除

添加

指定监控项完成后，手动修改表达式，为>200M，作为条件

问题事件生成模式 单个 多重

事件成功关闭 所有问题 所有问题如果标签值匹配

标记

标记	值	移除
----	---	----

[添加](#)

允许手动关闭 ☐

URL

描述

已启用 ☒

[添加](#) [取消](#)

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 关联项事件 自动发现 服务

触发器已添加

触发器

群组 所有 主机 agentzabbix.com

所有主机 / agentzabbix.com 已启用 ZBX SNMP JMX IPMI 应用集 11 监控项 46 触发器 21 图形 9 自动发现规则 2 Web 场景

过滤器 ▲

严重性 所有 未分类 信息 警告 一般严重 严重 灾难

状态 所有 正常 未知的

状态 所有 已启用 停用的

应用 重设

<input type="checkbox"/> 严重性	名称 ▲	表达式
<input type="checkbox"/> 警告	Template OS Linux: /etc/passwd has been changed on {HOSTNAME}	{agentzabbix.com.vfs.file.cksum[/etc/passwd].diff(0)}>0
<input type="checkbox"/> 警告	agent_memory>200M	{agentzabbix.com.memory_userd.last()}>200
<input type="checkbox"/> 信息	Template OS Linux: Configured max number of opened files is too low on {HOSTNAME}	{agentzabbix.com.kernel.maxfiles.last(0)}<1024

触发器创建完成后，创建监控显示的图形；

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 关联项事件 自动发现 服务

触发器已添加

触发器

群组 所有 主机 agentzabbix.com

所有主机 / agentzabbix.com 已启用 ZBX SNMP JMX IPMI 应用集 11 监控项 46 触发器 21 图形 9 自动发现规则 2 Web 场景

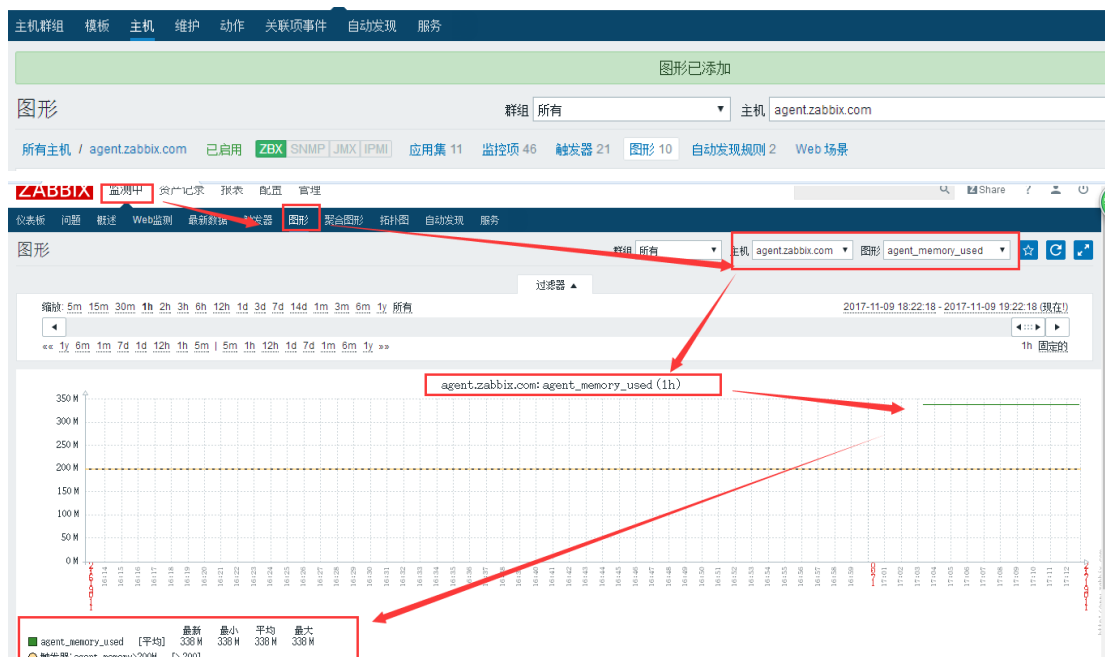
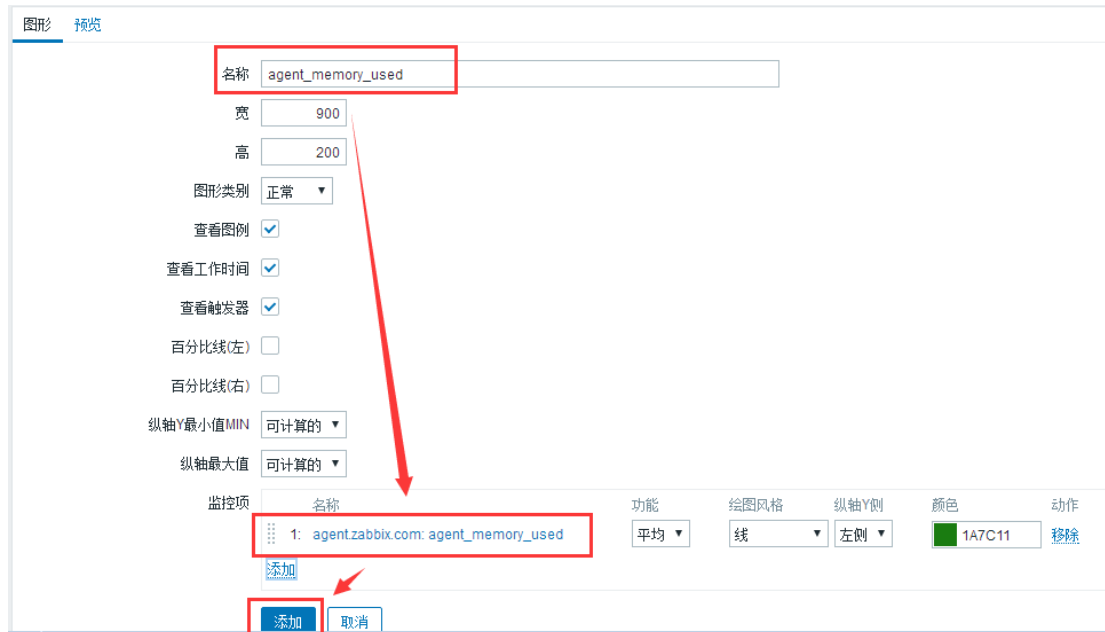
过滤器 ▲

严重性 所有 未分类 信息 警告 一般严重 严重 灾难

状态 所有 正常 未知的

状态 所有 已启用 停用的

应用 重设



打开首页，很明显的提示；



8.实现邮件报警；

首先配置 **server.zabbix.com** 端的邮件；

```
[root@server ~]# yum -y install postfix mailx dos2unix
```

```
[root@server ~]# mailx -V
```

```
[root@server ~]# vi /etc/mail.rc //在此文件末尾添加，指定接收邮件邮箱地址，指定邮
箱服务器地址，指定接收邮件邮箱地址的授权码，并非 163 邮箱的密码，而是授权码，登陆
163 网易邮箱地址后，--设置--开启 smtp 和 pop3--点击客户端生成授权码--使用此授权码进
行指定到此配置文件；
```

```
set from=li__wenhu@163.com smtp=smtp.163.com
```

```
set smtp-auth-user=li__wenhu@163.com smtp-auth-password=...
```

```
set smtp-auth=login
```

```
:wq
```

```
[root@server ~]# echo "测试内容" | mail -s "测试标题" li__wenhu@163.com //测试邮件
接收情况
```

```
[root@server ~]# vi /usr/local/zabbix/etc/zabbix_server.conf //打开 server 端主配置文
件的脚本存放位置
```

```
AlertScriptsPath=/usr/local/zabbix/share/zabbix/alertscripts
```

```
:wq
```

```
[root@server ~]# vi /usr/local/zabbix/share/zabbix/alertscripts/2.sh //编写报警邮件
```

脚本，调整 mail 命令语法位置，使用 dos2unix 命令转换字符，避免收到邮件不显示正常邮件内容而出现 ATT00001.bin 的错误；

```
#!/bin/bash
#export.UTF-8
bt=$2
sjr=$1
FILE=/tmp/mailtmp.txt
echo "$3" >$FILE
/usr/bin/dos2unix -k $FILE
/bin/mail -s "$bt" "$sjr" <$FILE
:wq
```

```
[root@server ~]# chmod 777 /usr/local/zabbix/share/zabbix/alertscripts/2.sh
[root@server ~]# chown zabbix:zabbix /usr/local/zabbix/share/zabbix/alertscripts/2.sh
[root@server ~]# touch /tmp/mailtmp.txt
[root@server ~]# chmod 777 /tmp/mailtmp.txt
[root@server ~]# chown zabbix:zabbix /tmp/mailtmp.txt //注意文件以及报警脚本
的归属以及权限，注：脚本文件和存放邮件消息文件归属必须是 zabbix；
[root@server ~]# /usr/local/zabbix/share/zabbix/alertscripts/2.sh li__wenhu@163.com "测试标
题 2" "测试内容 2" //测试脚本邮件接收情况
dos2unix: converting file /tmp/mailtmp.txt to Unix format ...
```

配置 zabbix web 网站进行监控项的邮件报警，每一张图片后附带解释和注意事项，大体步骤分为 监控--创建报警媒介类型--更新用户使用的报警媒介类型--创建动作（根据触发器触发）--验证；

创建报警媒介类型：



输入创建的报警媒介类型的名称，指定类型为脚本的方式，并且输入 zabbix_server 端 /usr/local/zabbix/share/zabbix/alertscripts 目录下的脚本名称，确保脚本的归属是 zabbix，并且脚本的权限是 777，下边设置脚本参数，也就是发送邮件的语法：脚本 收件人 标题 内容，必须严格按照此标准填写；

ZABBIX 监测中 资产记录 报表 配置 管理

一般 agent代理程序 认证 用户群组 用户 报警媒介类型 脚本 队列

报警媒介类型

名称 脚本邮件报警

类型 脚本

脚本名称 2.sh **zabbix_server脚本名**

脚本参数

参数

动作

{ALERT.SENDTO}

脚本后\$1, 收件人

移除

{ALERT.SUBJECT}

脚本后\$2, 标题

移除

{ALERT.MESSAGE}

脚本后\$3, 内容

移除

添加

已启用 ☒

添加 取消

创建报警媒介类型成功后，进行确认：

ZABBIX 监测中 资产记录 报表 配置 管理

一般 agent代理程序 认证 用户群组 用户 报警媒介类型 脚本 队列

报警媒介类型已添加

报警媒介类型

过滤器 ▲

名称 状态 任何 已启用 停用的

应用 重设

<input type="checkbox"/> 名称 ▲	类型	状态	用于动作中	细节
<input type="checkbox"/> Email	电子邮件	已启用		SMTP服务器: "mail.company.com", SMTP HELO: "company.com", SMTP电邮: "zabbix@company.com"
<input type="checkbox"/> Jabber	Jabber	已启用		Jabber 标识符: "jabber@company.com"
<input type="checkbox"/> SMS	短信	已启用		GSM 调制解调器: "/dev/ttyS0"
<input type="checkbox"/> 脚本邮件报警	脚本	已启用		脚本名称: "2.sh"

配置 zabbix 中的用户所使用的报警媒介类型以及接收邮件的邮箱；

ZABBIX 监测中 资产记录 报表 配置 管理

一般 agent代理程序 认证 用户群组 用户 报警媒介类型 脚本 队列

用户

过滤器 ▲

别名 名称 姓氏 用户

应用 重设

<input type="checkbox"/> 别名 ▲	用户名第一部分	姓氏	用户类型	群组	是否在线?
<input type="checkbox"/> Admin	Zabbix	Administrator	超级管理员	Zabbix administrators	是 (2017-07-11 20:09:3)
<input type="checkbox"/> guest			用户	Guests	不
<input type="checkbox"/> linux			用户	Zabbix administrators	不

选择报警媒介类型为刚刚创建的类型，并且输入接收邮件的地址，指定报警时间和报警级别；



创建动作，也就是发送邮件的动作；



输入动作名，动作名最好为英文，可以选择触发这个动作的条件（可以选择触发器=触发器的名称或者选择触发器似触发器名称），作为条件，如若如图不选择触发器的条件，那么任何消息都会提示到邮箱；

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
关联项事件
自动发现
服务

动作

动作

操作

恢复操作

确认操作

名称

agent_check

条件

标签

名称

A

维护状态 非在 维护

动作

移除

新的触发条件

触发器名称

似

添加

已启用

☒

添加

取消

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
关联项事件
自动发现
服务

动作

动作

操作

恢复操作

确认操作

名称

agent free mem >400

条件

标签

名称

A

触发器 = agent.zabbix.com: Agent Mem free >400

动作

移除

新的触发条件

触发器

=

在此输入搜索

选择

添加

已启用

☒

添加

取消

添加报警动作，一旦此触发器添加触发，添加报警邮件的标题和内容，默认操作步骤持续时间（修改持续时间为 60 秒（一分钟发送一次邮件）），添加操作（添加接收邮件的用户）；

默认接收人内容：故障{TRIGGER.STATUS},服务器:{HOSTNAME1}发生: {TRIGGER.NAME}故障!

默认信息内容：

告警主机:{HOSTNAME1}

告警时间:{EVENT.DATE} {EVENT.TIME}

告警等级:{TRIGGER.SEVERITY}

告警信息: {TRIGGER.NAME}

告警项目:{TRIGGER.KEY1}

问题详情:{ITEM.NAME}:{ITEM.VALUE}

当前状态:{TRIGGER.STATUS}:{ITEM.VALUE1}

事件 ID:{EVENT.ID}

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 关联项事件 自动发现 服务

动作

动作 操作 恢复操作 确认操作

默认操作步骤持续时间

60s

邮件发送间隔时间

默认接收人

故障(TRIGGER.STATUS).服务器:{HOSTNAME1}发生:{TRIGGER.NAME}故障!

邮件标题

默认信息

告警主机:{HOSTNAME1}

告警时间:{EVENT.DATE} {EVENT.TIME}

告警等级:{TRIGGER.SEVERITY}

告警信息:{TRIGGER.NAME}

邮件内容

维护期间暂停操作

☒

操作

步骤

细节

开始于

持续时间

动作

新的

添加

取消

添加操作（添加接收邮件的用户），步骤（指定邮件发送的次数），仅送到（选择创建的媒介类型）；

发送到用户

用户

动作

Admin (Zabbix Administrator)

移除

添加

仅送到

脚本邮件报警

默认信息

☒

条件

标签

名称

动作

新的

添加

取消

添加

取消

添加恢复操作（指故障恢复之后发送邮件的动作），具体步骤和添加操作相同；

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 关联项事件 自动发现 服务

动作

动作 操作 恢复操作 确认操作

与刚才创建的操作内容相同，当监控项恢复时，所发送恢复邮件提醒

默认接收人 故障 (TRIGGER.STATUS). 服务器: {HOSTNAME1} 发生: {TRIGGER.NAME} 故障!

默认信息
告警主机: {HOSTNAME1}
告警时间: {EVENT.DATE} {EVENT.TIME}
告警等级: {TRIGGER.SEVERITY}
告警信息: {TRIGGER.NAME}

操作 细节 动作

发送消息给用户: Admin (Zabbix Administrator) 通过 脚本邮件报警 编辑 移除

新的

添加 取消

确认动作创建的完成;

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 关联项事件 自动发现 服务

动作

动作 操作 恢复操作 确认操作

如上相同

默认接收人 故障 (TRIGGER.STATUS). 服务器: {HOSTNAME1} 发生: {TRIGGER.NAME} 故障!

默认信息
告警主机: {HOSTNAME1}
告警时间: {EVENT.DATE} {EVENT.TIME}
告警等级: {TRIGGER.SEVERITY}
告警信息: {TRIGGER.NAME}

操作 细节 动作

发送消息给用户: Admin (Zabbix Administrator) 通过 脚本邮件报警 编辑 移除

新的

添加 取消

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 关联项事件 自动发现 服务

动作

事件源 触发器 创建动作

过滤器

名称 状态 任何 已启用 停用的

应用 重设

名称	条件	操作	状态
<input type="checkbox"/> agent_check		发送消息给用户: Admin (Zabbix Administrator) 通过 脚本邮件报警 发送消息给用户群组: Zabbix administrators 通过 脚本邮件报警	已启用
<input type="checkbox"/> Report problems to Zabbix administrators		发送消息给用户群组: Zabbix administrators 通过 所有介桥	已启用

显示 已自动发现的 2 中的 2

开始测试邮件报警

测试一: 首先关闭 agent.zabbix.com 主机的 ftp 服务, 测试邮件报警情况;

[root@agent ~]# /etc/init.d/vsftpd stop

关闭 vsftpd:

[确定]

Dashboard

添加仪表盘 / Dashboard

常用的图形

未添加数据图.

已更新: 06:47:12

常用的聚合图形

未添加聚合图形.

已更新: 06:47:12

常用的拓扑图

未添加拓扑图.

已更新: 06:47:12

问题

时间 ▼	恢复时间	状态	信息	主机	问题·严重性	持续时间
06:43:30				win2k8dc1	Zabbix agent on win2k8dc1 is unreachable for 5 minutes	3m 42s
06:36:58				agent.zabbix.com	FTP service is down on agent.zabbix.com	10m 14s
06:28:58	06:33:58			agent.zabbix.com	FTP service is down on agent.zabbix.com	5m

ID 完成

2017-11-10 06:37:02	agent_check	脚本邮件报警	Admin (Zabbix Administrator) li_wenhu@163.com	主题: 故障PROBLEM.服务器:agent.zabbix.com发生: FTP service is down on agent.zabbix.com故障!
				消息: 告警主机:agent.zabbix.com
				告警时间:2017.11.10 06:36:58
				告警等级:Average
				告警信息: FTP service is down on agent.zabbix.com
				告警项目:net.tcp.service[ftp]
				问题详情:FTP service is running:Down (0)
				当前状态:PROBLEM:Down (0)
				事件ID:136

163 网易免费邮 mail.163.com 严选 li_wenhu@163.com 易信 专业版 升级VIP 升级服务 设置 帮助 退出

首页 通讯录 应用中心 收件箱 一起拼 抢新 网易严选 入店 Problem: Za... 故障PROBLE...

收信 写信

收件箱 红旗邮件 待办邮件 星标联系人邮件 草稿箱 已发送 其他2个文件夹 邮件标签 邮箱中心 文件中心 附件图集

网选严选11.11 秋冬新品发布 买1送1 全场S折+红包 再送iPhoneX >

故障PROBLEM,服务器:agent.zabbix.com发生: FTP service is down on agent.zabbix.com故障!

发件人: 我<li_wenhu@163.com> 收件人: 我<li_wenhu@163.com> 时间: 2017年11月09日 22:44 (星期四)

【通知】你有一张双11全场折扣卡待领! 立即领取>>

告警主机:agent.zabbix.com

告警时间:2017.11.10 06:36:58

告警等级:Average

告警信息: FTP service is down on agent.zabbix.com

告警项目:net.tcp.service[ftp]

问题详情:FTP service is running:Down (0)

当前状态:PROBLEM:Down (0)

测试二：关闭 win2k8 主机，测试邮件报警情况；

win2k8dc1

打开此虚拟机电源 编辑虚拟机设置

设备

内存	1 GB
处理器	1
硬盘 (SCSI)	60 GB
CD/DVD (IDE)	使用文件 E:\系...
网络适配器	自定义 (VMnet1)
显示器	自动检测

描述

为该虚拟机在此处输入描述:

Dashboard

添加仪表盘 / Dashboard

常用的图形

未添加数据图.

已更新: 06:49:39

常用的聚合图形

未添加聚合图形.

已更新: 06:49:39

常用的拓扑图

未添加拓扑图.

已更新: 06:49:39

问题

时间	恢复时间	状态	信息	主机	问题·严重性	持续时间
06:43:30		问题		win2k8dc1	Zabbix agent on win2k8dc1 is unreachable for 5 minutes	6m 10s
06:36:58		问题		agent.zabbix.com	FTP service is down on agent.zabbix.com	12m 42s
06:28:58	06:33:58	已解		agent.zabbix.com	FTP service is down	5m

动作日志

过滤器 ▲

接收者 选择

应用

重设

缩放: 所有

◀ ◀

...

« »

时间	动作	类型	接收者	消息
2017-11-10 06:43:32	agent_check	脚本邮件报警	Admin (Zabbix Administrator) li_wenhu@163.com	主题: 故障PROBLEM.服务器:win2k8dc1发生: Zabbix 消息: 告警主机:win2k8dc1 告警时间:2017.11.10 06:43:30 告警等级:Average 告警信息: Zabbix agent on win2k8dc1 is unrea 告警项目:agent.ping 问题详情:Agent ping:Up (1)

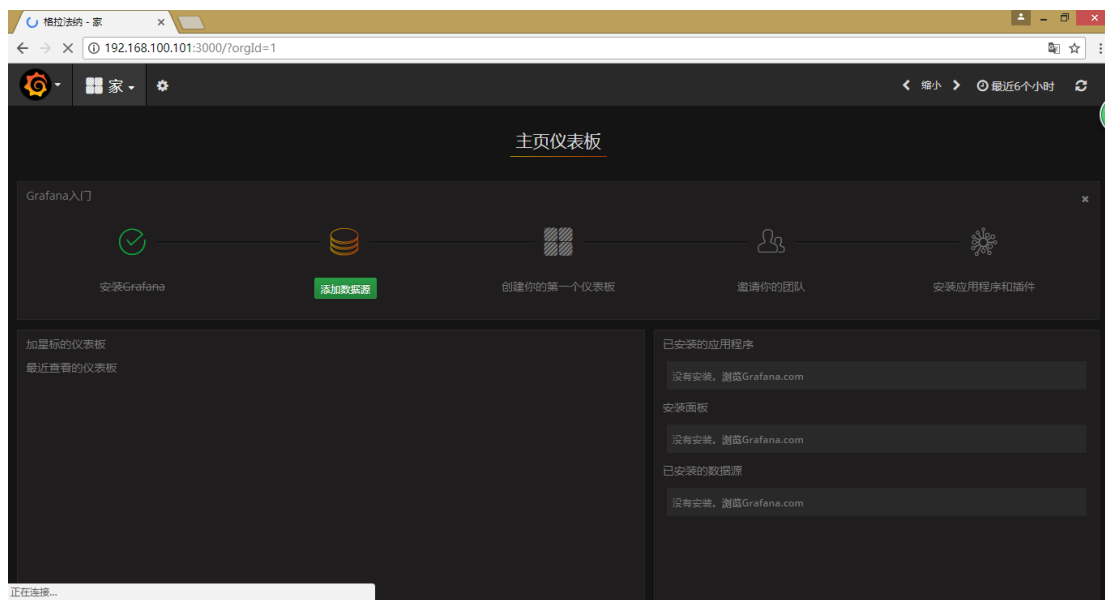


9.安装并配置 Grafana;

```
[root@server ~]# wget https://s3-us-west-2.amazonaws.com/grafana-releases/release/grafana-4.6.1-1.x86_64.rpm
[root@server ~]# yum -y localinstall grafana-4.6.1-1.x86_64.rpm
[root@server ~]# grafana-cli plugins install alexanderzobnin-zabbix-app ##安装 zabbix 插件
[root@server ~]# mkdir -p /opt/grafana/data/plugins/
[root@server ~]# cp -r /var/lib/grafana/plugins/alexanderzobnin-zabbix-app/ /opt/grafana/data/plugins/
[root@server ~]# ls /opt/grafana/data/plugins/
alexanderzobnin-zabbix-app
[root@server ~]# chmod 777 /opt/grafana/data/plugins/ -R
[root@server ~]# systemctl start grafana-server
[root@server ~]# netstat -tln |grep gra
Proto Recv-Q Send-Q Local Address          Foreign Address         State
PID/Program name
tcp        0      0 0.0.0.0:3000            0.0.0.0:*               LISTEN
57103/grafana-serve
```

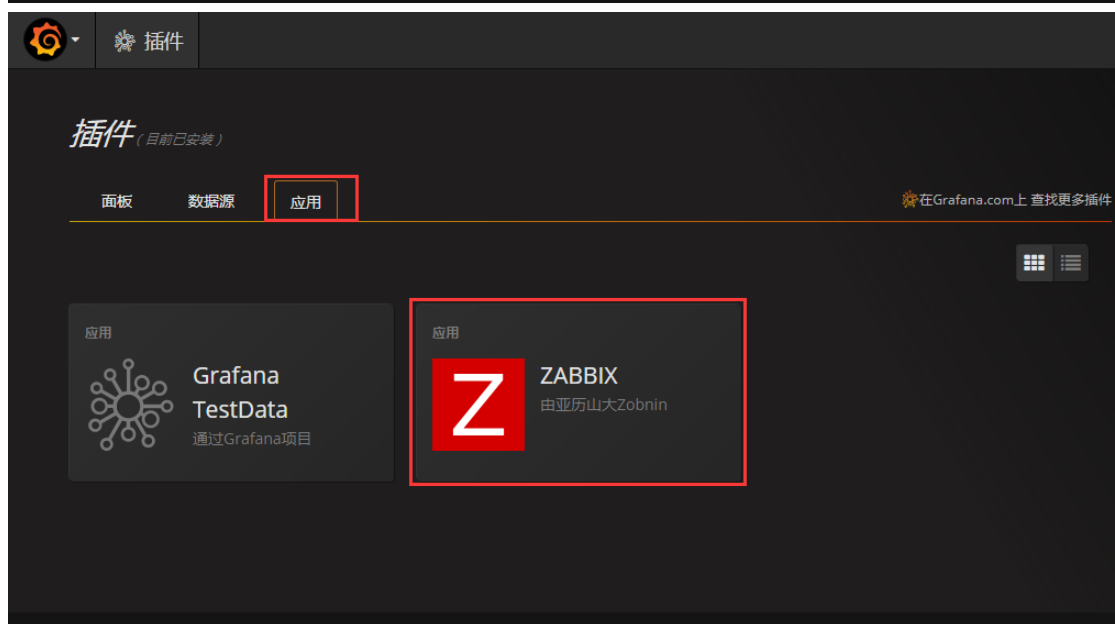
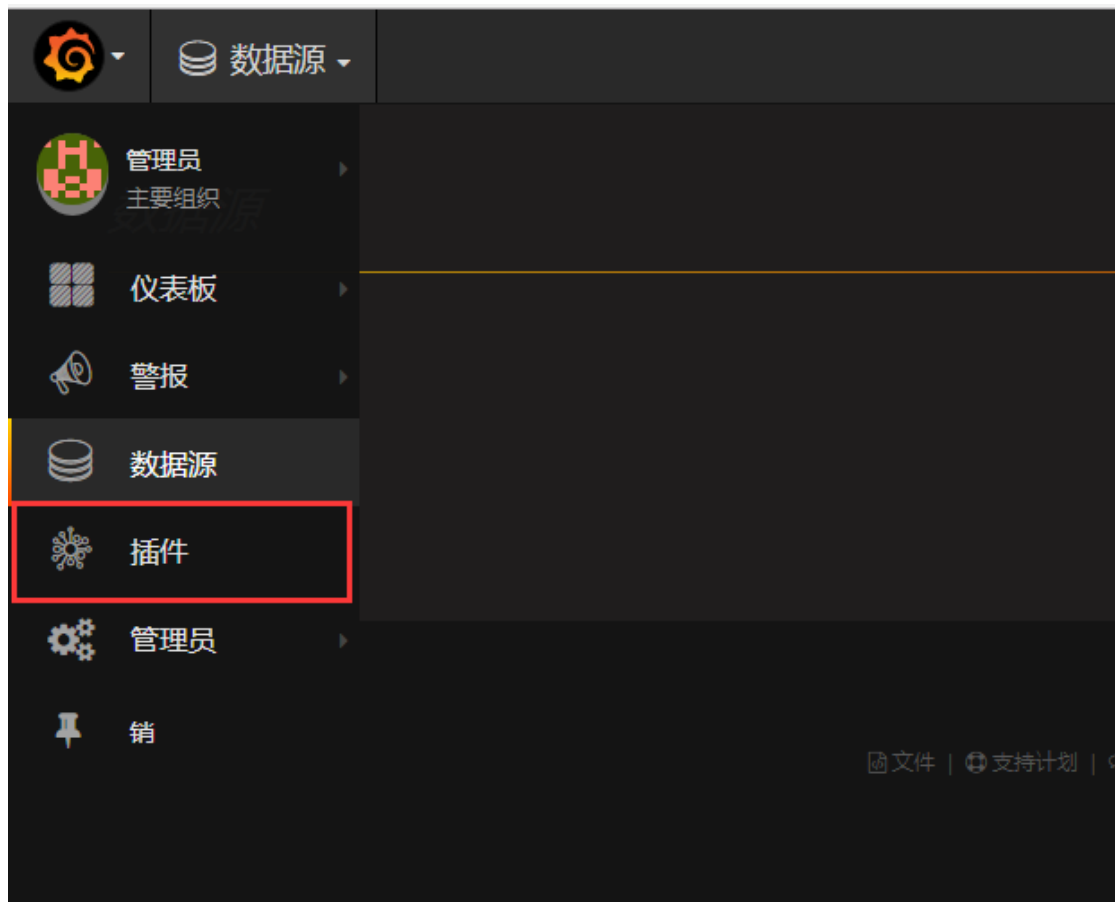
访问测试 <http://192.168.100.101:3000>, 默认用户名 **admin**, 密码 **admin**, 在此是谷歌浏览器翻译过后的界面, 大家可以自行下载谷歌浏览器, 下载翻译插件, 在此不做过多介绍; 安装 **grafana**;

http://192.168.100.101:3000/login



10.配置 Grafana 的 web 界面，呈现效果；

启用插件；





插件



ZABBIX

通过

应用

配置

仪表板

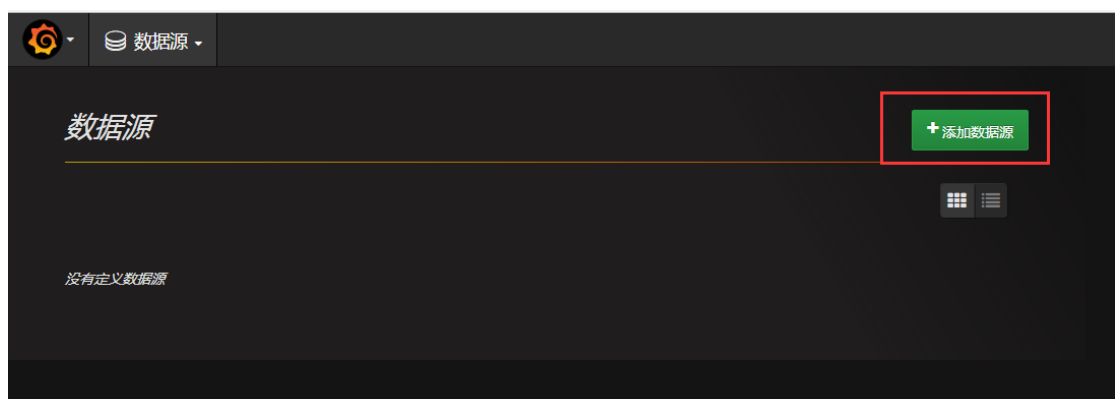
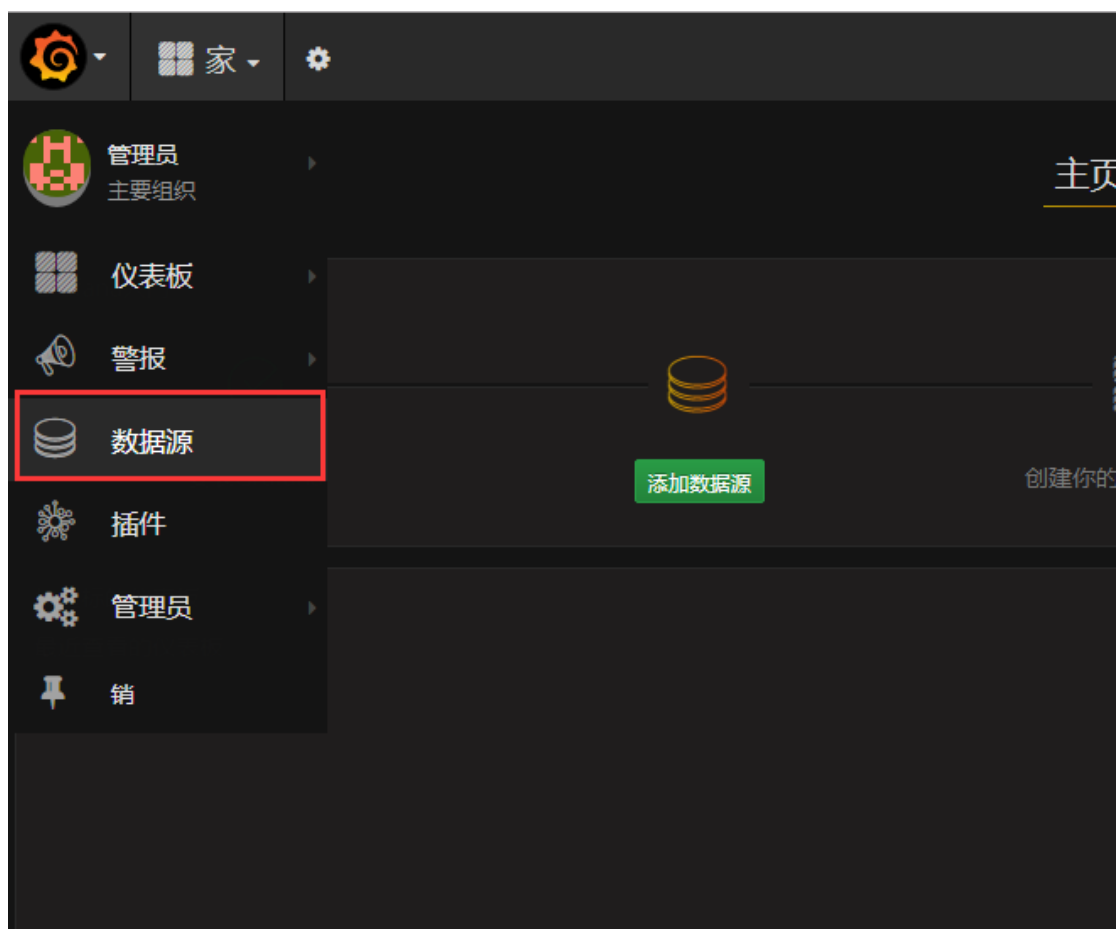
读我

Zabbix插件配置

启用



添加数据源；





数据源

添加数据源

配置

仪表板

名称

zabbix

?

默认

☒

类型

Zabbix

▼

网址

http://192.168.100.101/zabbix/api_jsonrpc..?

?

访问

direct

▼

?

api_jsonrpc.php

基本身份验证

☐

凭证

?

☐

Zabbix API的细节

用户名

admin

密码

.....

趋势

☐

缓存TTL

1h

此处为登陆zabbix 的web界面的用户密码，密码为zabbix

直接数据库连接

启用

☐

警报

启用警报

☐

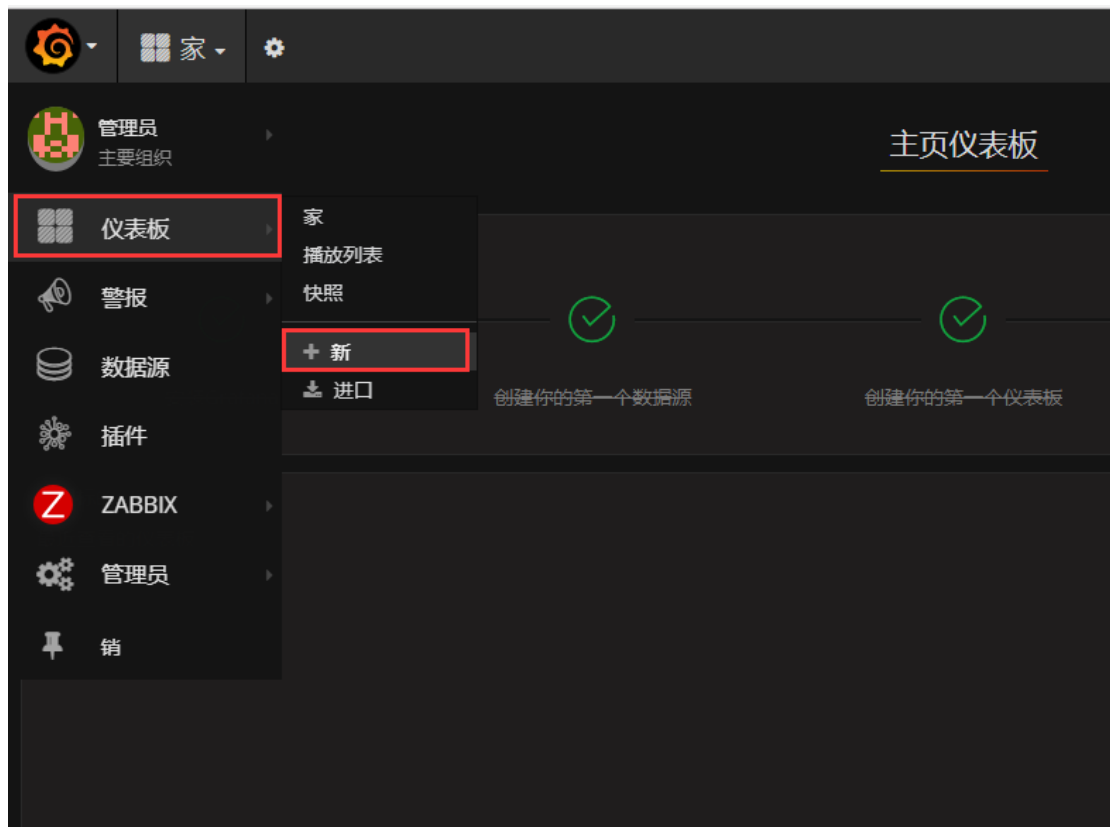
✓ Zabbix API版本 : 3.4.2

保存并测试

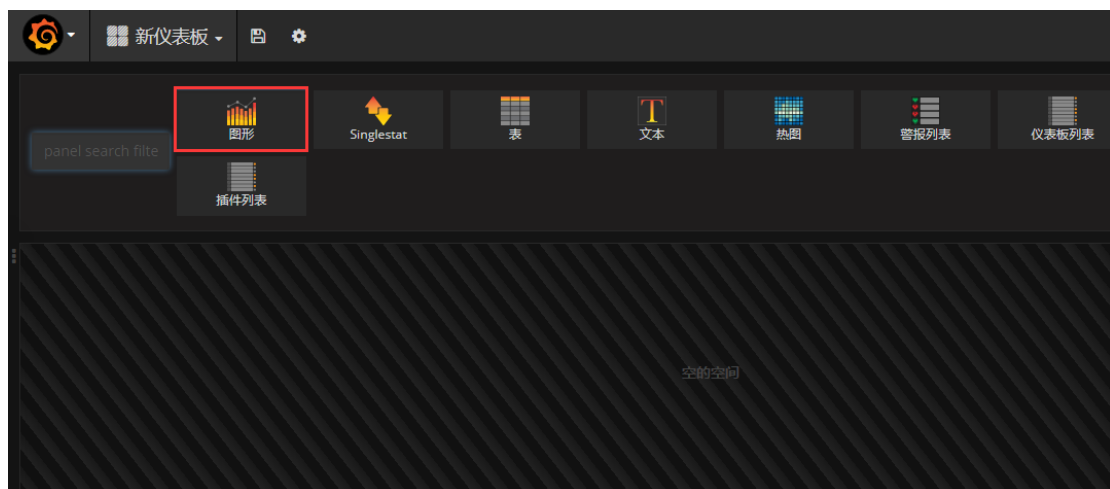
删除

取消

创建仪表盘；



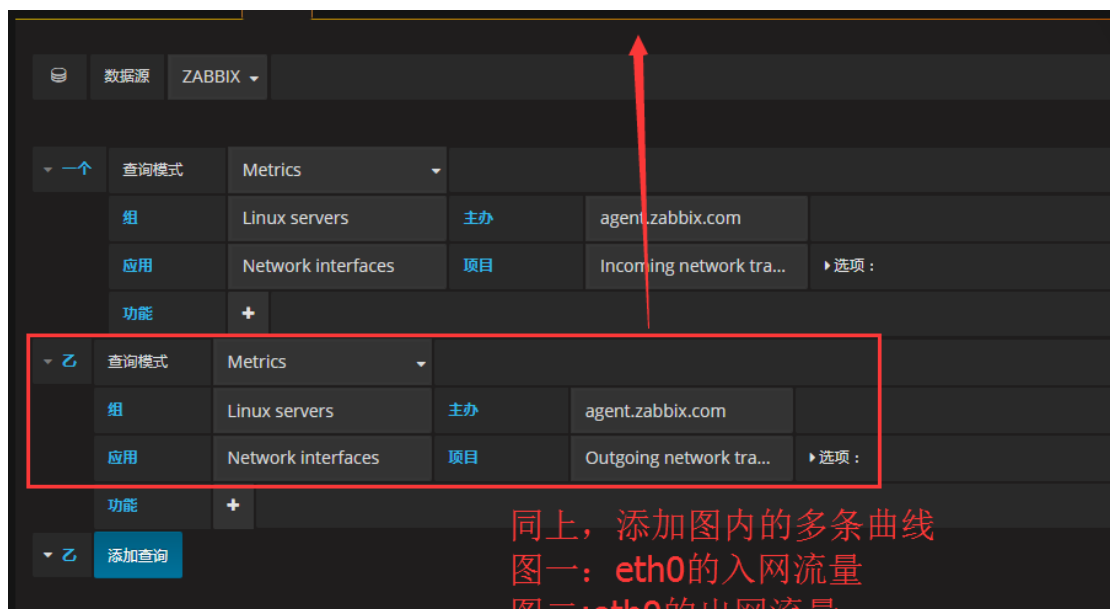
创建仪表盘时指定显示的图形；



输入图形名称；



选择数据源和图中需要显示的主机和主机的监控项；

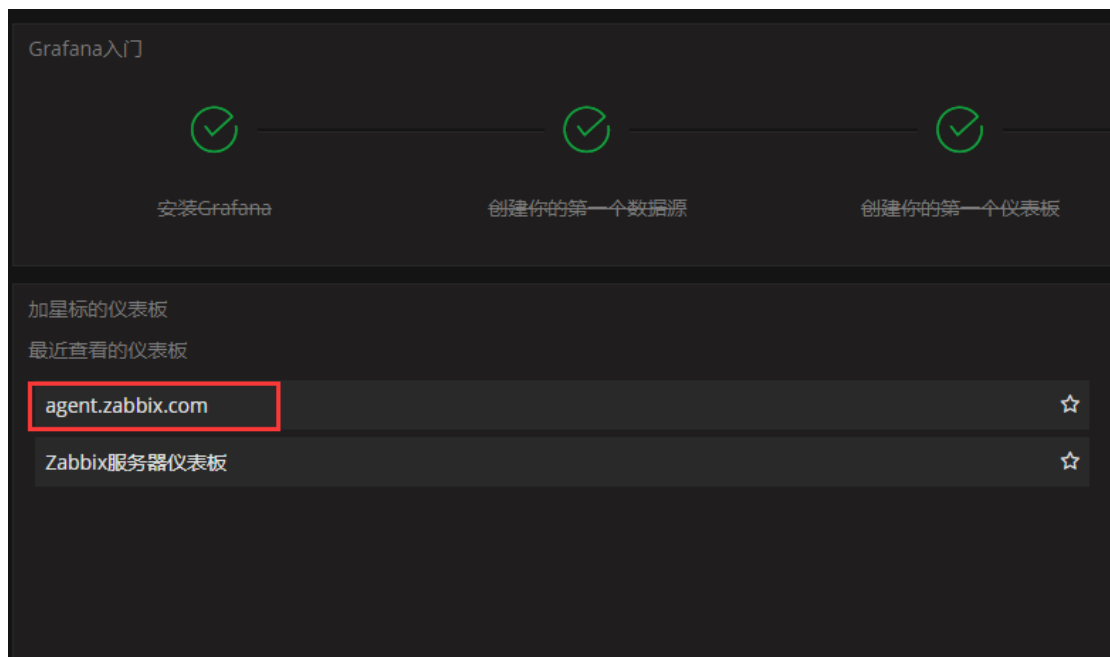
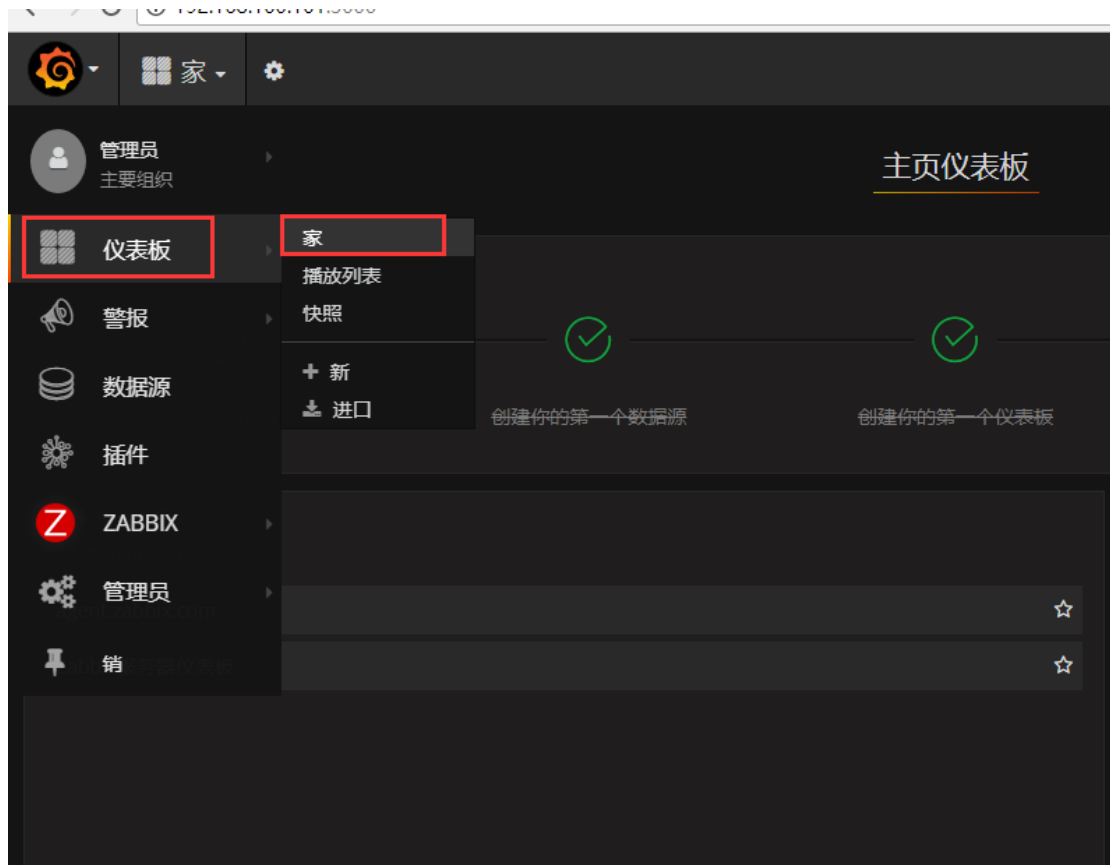


创建完成，保存此仪表盘，输入仪表盘的名称；

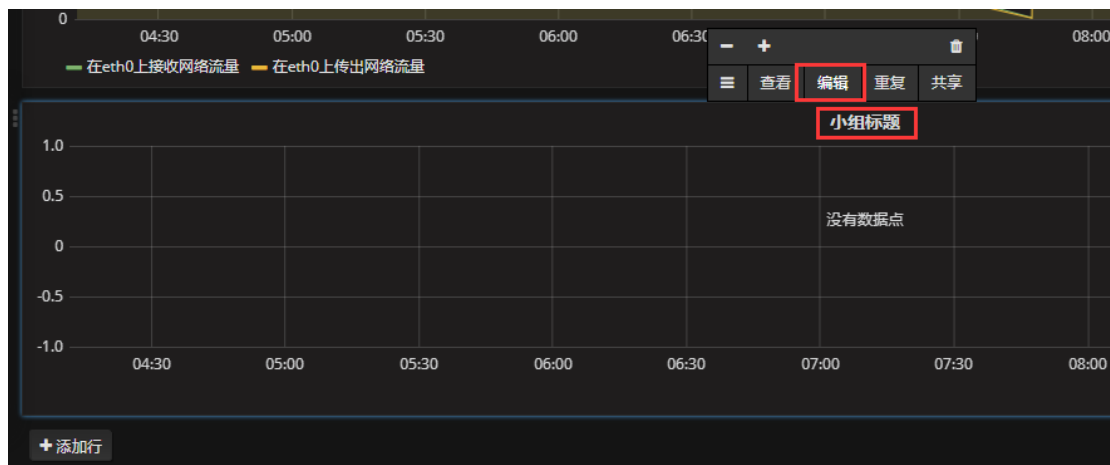
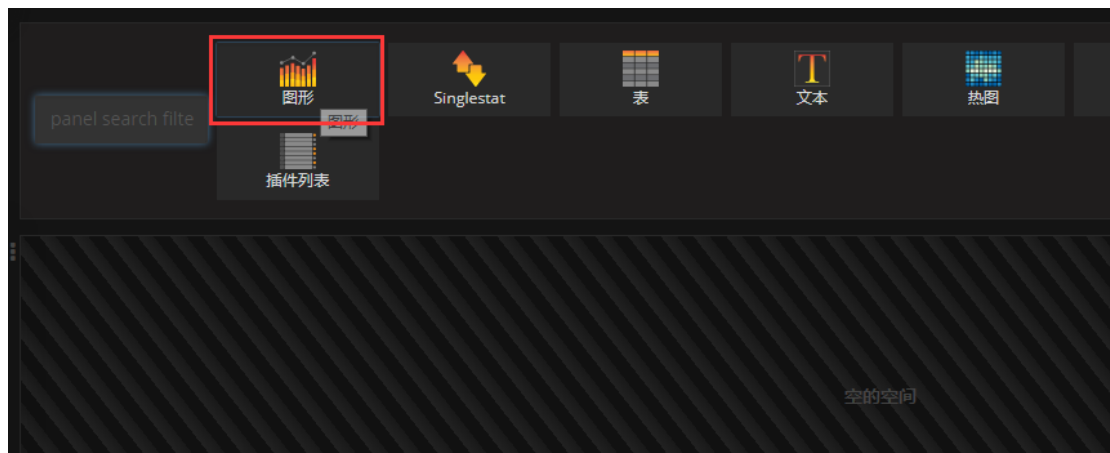


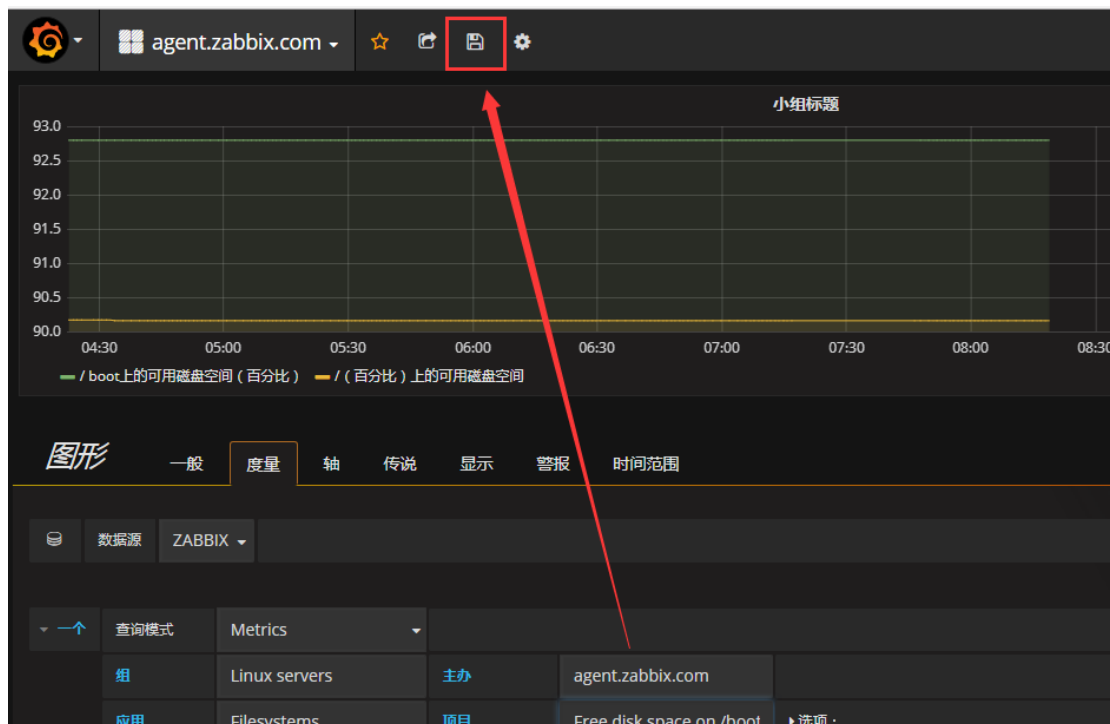
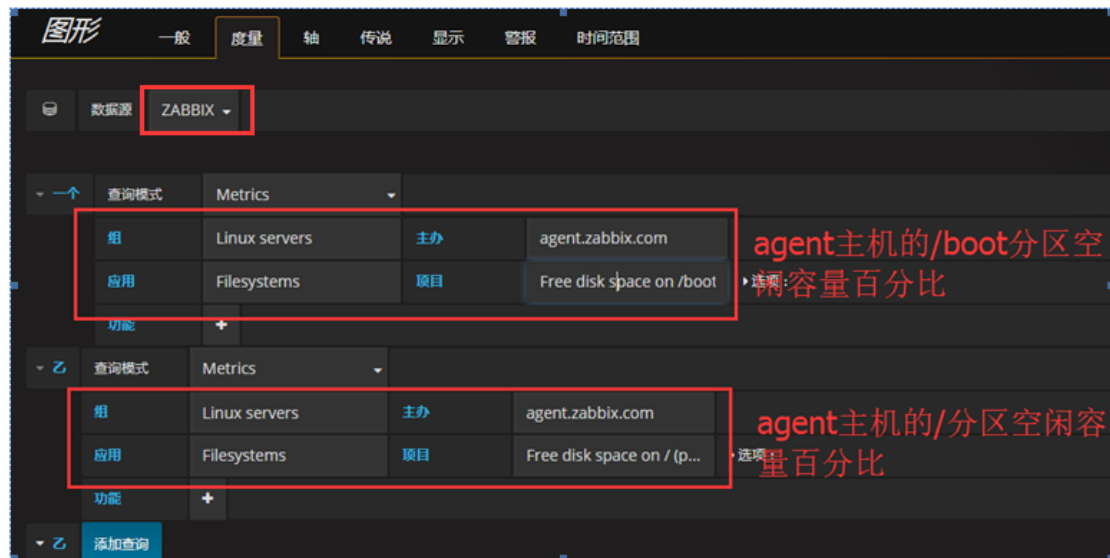
The image shows a '另存为...' (Save As...) dialog box in Zabbix. The title bar contains a document icon and the text '另存为...'. The main area has a large red text prompt: '此名称为dashboard的名称' (This name is the name of the dashboard). Below this, there is a text input field labeled '新的名字' (New Name) containing the text 'agent.zabbix.com'. At the bottom, there are two buttons: '保存' (Save) and '取消' (Cancel).

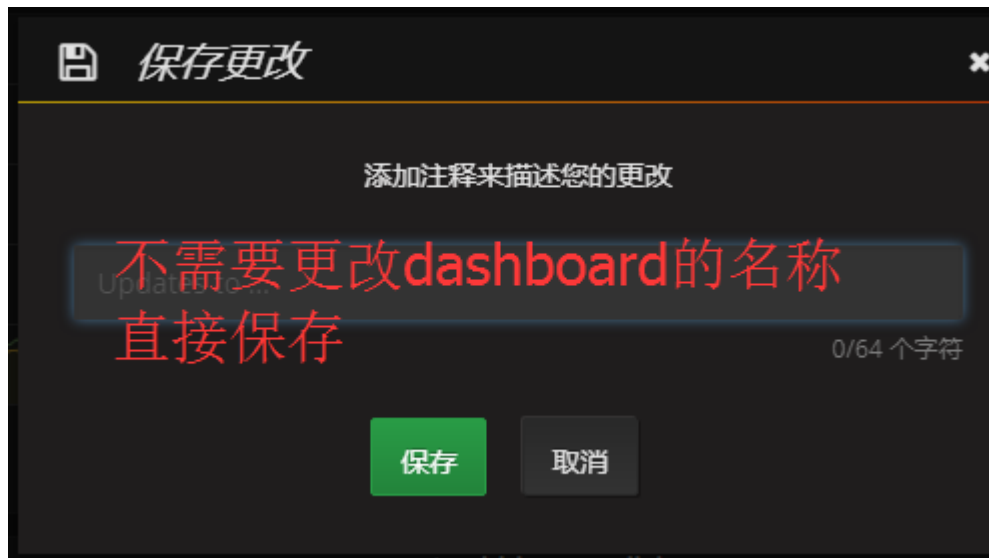
验证创建的仪表盘；



在现有的 **dashboard** 中添加图形；







验证最终的仪表盘显示图形；



四.项目实验总结（遇到的错误及解决方案，难点、重点解说，扩展学习）

1.错误及解决方案：

错误一：监控 Mysql 数据库时，没有图形；

错误及解决方案：P/13-18

错误二：配置报警邮件时，邮箱收到的邮件为 ATT00001.bin 的内容；

错误及解决方案：P/38

邮箱收到邮件出现 ATT00001.bin 的内容：

原因：zabbix 无法将内容转义成中英文，需要借助 dos2unix 工具进行转换

2.重点、难点：

3.扩展学习：