**Zabbix服务安装部署及监控配置**

第一节

官方网站：http://www.zabbix.com/

The Enterprise-class Monitoring Solution for Everyone

企业级监控解决方案

zabbix 是一个基于 WEB 界面的提供分布式系统监视以及网络监视功能的企业级的开源解决方案

zabbix 能监视各种网络参数，保证服务器系统的安全运营；并提供灵活的通知机制以让系统管理员快速定位解决存在的各种问题。

# Zabbix 主要特点

安装与配置简单，学习成本低 支持多语言（包括中文） 免费开源 自动发现服务器与网络设备分布式监视以及 WEB 集中管理功能 可以无 agent 监视 用户安全认证和柔软的授权方式 通过 WEB 界面设置或查看监视结果 email 等通知功能

# Zabbix主要功能

监控CPU负载、内存使用、磁盘使用、网络状态、端口监视、日志监视、插件开发自定义

# Zabbix运行机制

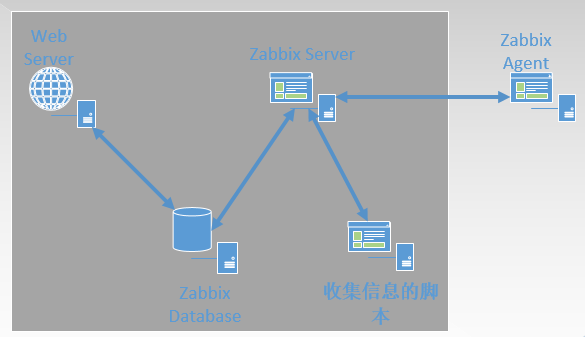
Zabbix Server 和 Zabbix agent

Zabbix Server 可以通过 SNMP，Zabbix agent，Ping，端口监视等你方法提供对远程服务器/网络状态的 监视，数据收集功能，可以在运行 Linux，Solaris，HP-UX，AIX，FreeBSD，OpenBSD，OX X，Windows 多 平台。

Zabbix Agent 安装在需要被监控的目标服务器上，主要完成对硬件信息与操作系统有关的内存，CPU 等你 信息收集。

Zabbix Server 可以单独监视远程服务器的服务状态，同时也可以与 Zabbix Agent 结合。可以轮询 Zabbix Agent 主动接收监视数据（trapping 传递 方式），同时还可以被动接收 Zabbix Agent 发送的数据。

# Zabbix运行过程图



Zabbix 支持多种网络方式下的监控，可以通过分布式的部署和安装监控代理 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！Agent 监控端口 10050

服务端安装 Zabbix Server 和 Zabbix Agent 两个服务，客户端只安装 Zabbix Agent 一个服务，监控原理 就是 Zabbix Agent 服务将从系统中获取的相关数据传送给 Zabbix Server，然后 Zabbix Server 服务奖数据分 析之后以 PHP 页面展示出来！

整个环境所需要的软件包

php-5.6.13

libmcrypt-2.5.8

mysql-5.6.26

nginx-1.8.0

zabbix-3.4.3

我们需要先构建 LNMP 的环境，因为 Zabbix 通过 PHP 页面展示

搭建 LNMP 的环境，如果有这个环境可以忽略

创建导入zabbix数据库

Zabbix 需要数据库的支持，而却 Zabbix 已经自带了数据库结构，images 数据库和表数据等相关内容

创建 Zabbix 使用数据库

[root@xuegod63 ~]# mysql -uroot -p123456

mysql> create database zabbix;

mysql> grant all on zabbix.\* to zabbix@xuegod63 identified by 'zabbix'; #zabbix 用户授权

mysql> flush privileges;

mysql> exit

[root@xuegod63 ~]# tar xvf zabbix-3.4.3.gz -C /usr/local/src/ [root@xuegod63 ~]# cd /usr/local/src/ zabbix-3.4.3/

导入数据库，注意顺序，否则会报错

[root@xuegod63 zabbix-3.4.3]# mysql -uzabbix -pzabbix zabbix < database/mysql/schema.sql

[root@xuegod63 zabbix-3.4.3]# mysql -uzabbix -pzabbix zabbix < database/mysql/images.sql

[root@xuegod63 zabbix-3.4.3]# mysql -uzabbix -pzabbix zabbix < database/mysql/data.sql

如果在做分布式的环境中，Proxy 服务器只需要一个数据库 schema，代理者 agent 不需要数据库!Server 必须 三个数据库同时拥有

Zabbix同时支持很多类型数据库，可以通过查阅帮助： https://www.zabbix.com/documentation/2.4/manual/appendix/install/db\_scripts

编译安装 zabbix

[root@xuegod63 ~]# groupadd zabbix

[root@xuegod63 ~]#useradd -s /sbin/nologin -g zabbix zabbix#创建用户

[root@xuegod63zabbix-3.4.3]#

yum install -y net-snmp-devel

yum install libevent libevent-devel -y#解决依赖

[root@xuegod63zabbix-3.4.3]# ./configure --prefix=/usr/local/zabbix --enable-server --enable-agent --with-mysql=/var/lib/mysql/bin/mysql\_config--with-net-snmp --with-libcurl

参数说明

**--prefix= #指定路径**

**--enable-server #Server 支持**

**--enable-agent #支持 Zabbix 客户端**

**--with-mysql #指定 MySql 库可以选择自定路径 mysql\_config，mysql\_config 是命令，用于编译 mysql 客户端程序**

**--with-net-snmp #支持 snmp 协议，需要安装 net-snmp-devel 包**

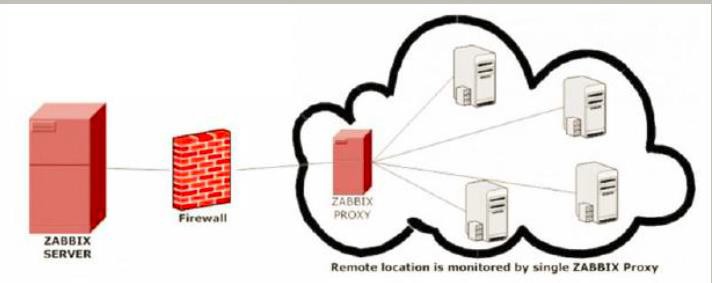
**--with-libcurl #支持 CURL 功能，libcurl 主要功能就是用不同的协议连接不同的服务器，**

**libcurl 当前支持的协议有 http，https，ftp，gopher，telent，dict，file，和 ldap 协议**

**其他参数：**

**--enable-proxy 指的是支持 zabbix 代理服务器，zabbix proxy 是一个监控代理服务器，它收集到监控数据，选存放在缓冲区，保存的时间可以通过配置文件设定，然后再传送的 zabbix server。监控代理服务需要一个独立的数据库。**

**用 zabbix proxy 的好处：进程监控，当监控的位置通信不便时，当通讯上千台设备的时候，使用 zabbix\_proxy 可以简化维护分布式监控，降低 zabbix server 的负负载，**



看到如下信息，预编译就没有什么问题

……

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Now run 'make install' \*

\* \*

\* Thank you for using Zabbix! \*

\* <http://www.zabbix.com> \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[root@xuegod63 zabbix-3.4.3]# make install

#按上面的提示，没有 make 步骤，直接 make install

[root@xuegod63 ~]# ls /usr/local/zabbix/ //编译成功，产生如下目录

bin etc lib sbin share

配置 zabbix Server

[root@xuegod63~]#vim/usr/local/zabbix/etc/zabbix\_server.conf

DBHost=xuegod63 #默认注释掉了，直接取消注释即可

DBName=zabbix #数据库用户，我们授权的用户也是zabbix

DBUser=zabbix #默认是 root，我们授权的用户是 zabbix

DBPassword=zabbix #密码我们授权的也是 zabbix

监控 Zabbix Server 本身

监控本身，意思是本身作为服务器之外，自己也做自己的客户端，也要使用agentd这个代理者 配置文件中，有 agentd 和 agent 两个配置文件，前者是守护进程，后者依赖 xinetd

[root@xuegod63~]#vim /usr/local/zabbix/etc/zabbix\_agentd.conf

Server=127.0.0.1 #默认监控服务器自己，这三行不用改

ServerActive=127.0.0.1

Hostname=Zabbix server

UnsafeUserParameters=1 #允许所有的字符是在用户定义的参数，参数传递，也就是支持自定义脚本

其中 Server 和 ServerActive 都指定 zabbixserver 的 IP 地址，不同的是，前者是被动后者是主动。也就是说前者允许 127.0.0.1 这个 ip 来我这取数据。而 serverActive 的 127.0.0.1 的意思是，客户端主动提交数据给他。

启动服务

如果 zabbix 用户不存在， 将以 root 运行，新建之后，如果 zabbix 存在，那么直接可以用 zabbix 运行

直接运行

[root@xuegod63 ~]#/usr/local/zabbix/sbin/zabbix\_server

sbin/zabbix\_server: error while loading shared libraries: libmysqlclient.so.20: cannot open shared object file: No such file or directory#如果启动报错解决

[root@xuegod63 zabbix]# vim /etc/ld.so.conf#解决库文件找不到

include ld.so.conf.d/\*.conf

/var/lib/mysql/lib/ #根据实际路径添加此行

[root@xuegod63 zabbix]#ldconfig#使库文件生效

[root@xuegod63~]#echo /usr/local/zabbix/sbin/zabbix\_server >> /etc/rc.local

#开机启动

[root@xuegod63 ~]# netstat -antup | grep zabbi

#server 运行端口 10051

tcp 0 0 0.0.0.0:10051 0.0.0.0:\* LISTEN37568/zabbix\_server

或者通过启动脚本启动 zabbix

Zabbix默认有部分启动脚本，如果你自己也可以通过脚本的方式来实现，可以使用他的模板来实现 脚本在源码目录的 misc/init.d/下面(使用 fedora 下面的 core 即可，也可以使用 fedroa core5 的脚本均可)

[root@xuegod63~]#cd /usr/local/src/zabbix-3.4.3/misc/init.d/

[root@xuegod63 init.d]# cp fedora/core/\* /etc/rc.d/init.d/

[root@xuegod63 init.d]# vim /etc/init.d/zabbix\_server

BASEDIR=/usr/local/zabbix #找到此行，并修改

[root@xuegod63 init.d]# vim /etc/init.d/zabbix\_agentd ； cd

BASEDIR=/usr/local/zabbix #找到此行，并修改

[root@localhost core]# vim /etc/rc.d/init.d/zabbix\_agentd

[root@localhost core]# /etc/init.d/zabbix\_agentd start

Reloading systemd: [ OK ]

Starting zabbix\_agentd (via systemctl): [ OK ]

[root@localhost core]# chkconfig zabbix\_server on

[root@localhost core]# chkconfig zabbix\_agentd on

编辑 php 页面控制文件

默认配置文件没有写入到我们的 Nagix 的配置文件中，也就是不能加载到我们的 zabbix 页面目录

[root@xuegod63~]#cp -r /usr/local/src/zabbix-3.4.3/frontends/php/\* /usr/local/nginx/html/

[root@xuegod63 ~]# vim /usr/local/nginx/conf/nginx.conf

location / {

root html;

index index.php index.html index.htm; #找到此行内容，添加index.php

}

重启nginx服务和php-fpm服务

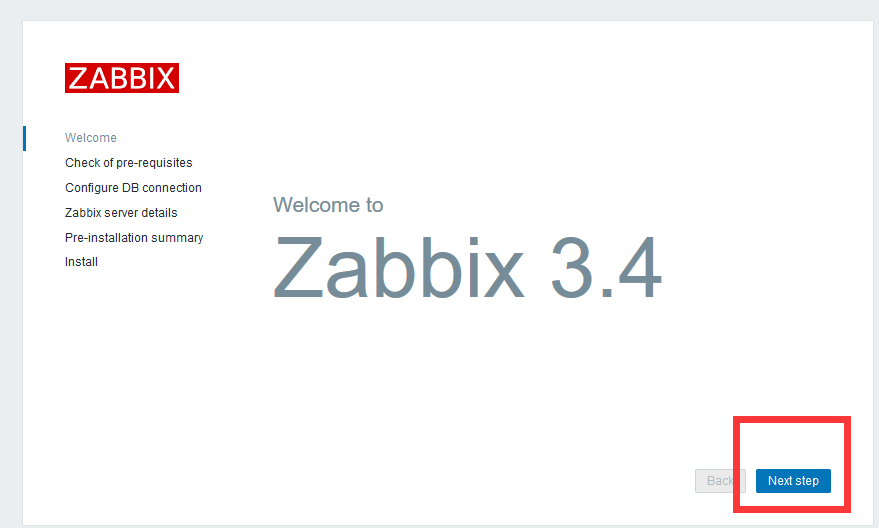
[root@xuegod63 ~]# /usr/local/nginx/sbin/nginx -s reload ;

[root@xuegod63 ~]#/etc/init.d/zabbix\_server restart

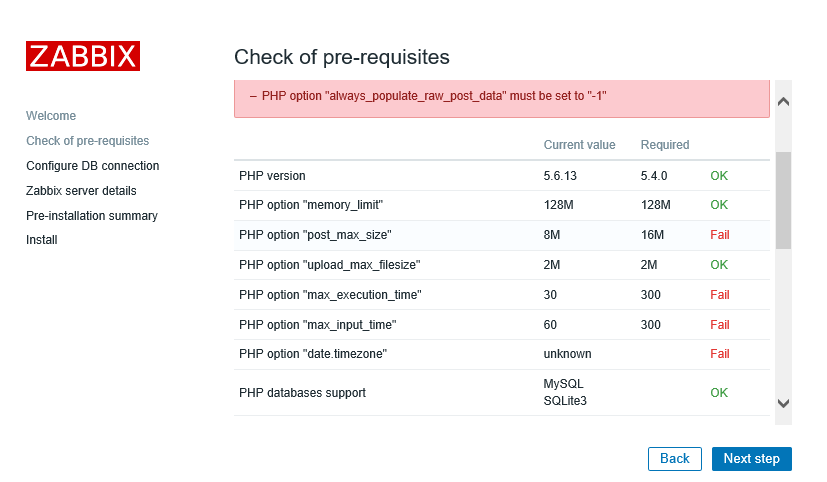
[root@xuegod63 ~]#/etc/init.d/php-fpm restart

打开网页安装 zabbix web

浏览器输入：<http://192.168.142.125> 如图，下一步 Next



开始检查环境，如图，发现很多环境不符合条件



编辑 php.ini 文件，修改环境

[root@xuegod63 ~]# vim /usr/local/php/php.ini

post\_max\_size = 16M #修改

max\_execution\_time = 300 #修改

max\_input\_time = 300 #修改

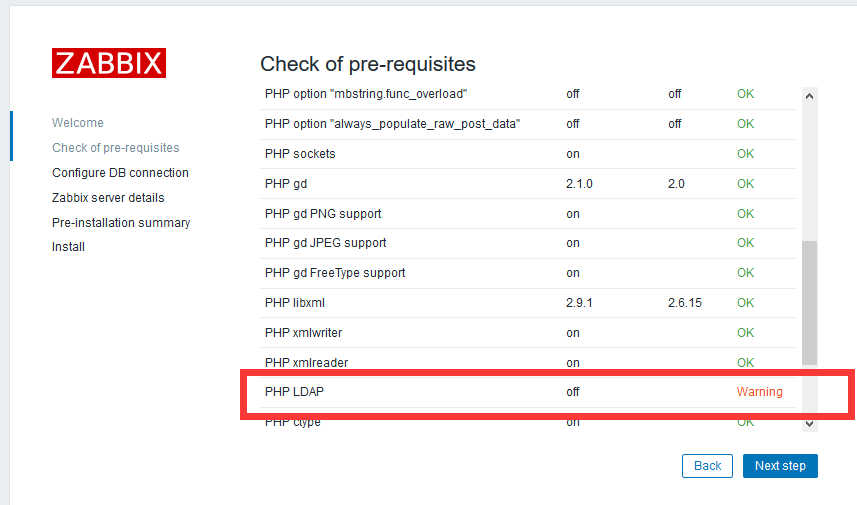
date.timezone = Asia/Shanghai #此行有注释，去注释并修改

always\_populate\_raw\_post\_data = -1 #此行取消注释即可

mysqli.default\_socket = /var/lib/mysql/mysql.sock #指定php连接mysql的sock路径

[root@xuegod63 ~]# /etc/init.d/php-fpm restart #重启fpm

刷新页面



**如果提示没有安装PHP LDAP模块，为了不影响后面的功能，现在把这个模块重新编译安装进去：**

**可以单独把这个模块编译安装进来：**

**[root@xuegod63zabbix-3.4.3]# cd /usr/local/src/php-5.6.13/ext/**

**[root@xuegod63 ext]# cd ldap/**

**[root@xuegod63 ldap]# /usr/local/php/bin/phpize**

**Configuring for:**

**PHP Api Version: 20131106**

**Zend Module Api No: 20131226**

**Zend Extension Api No: 220131226**

**[root@xuegod63 ldap]# ./configure --with-php-config=/usr/local/php/bin/php-config --with-ldap**

**上面的过程会报错,提示缺少库文件:**

**解决:**

**[root@xuegod63 ~]# yum install openldap-devel openldap -y**

**[root@xuegod63 ~]# rsync -avz /usr/lib64/libldap\* /usr/lib/**

**再次configure,即可通过预编译**

**[root@xuegod63 ldap]# make && make install**

**[root@xuegod63 ~]# vim /usr/local/php/php.ini**

**添加：extension=ldap.so**

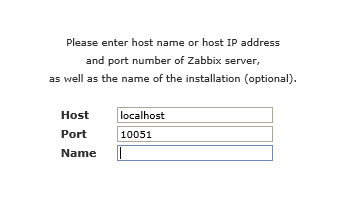
**[root@xuegod63 ~]# /etc/init.d/php-fpm restart**

**再次刷新即可**

下一步：

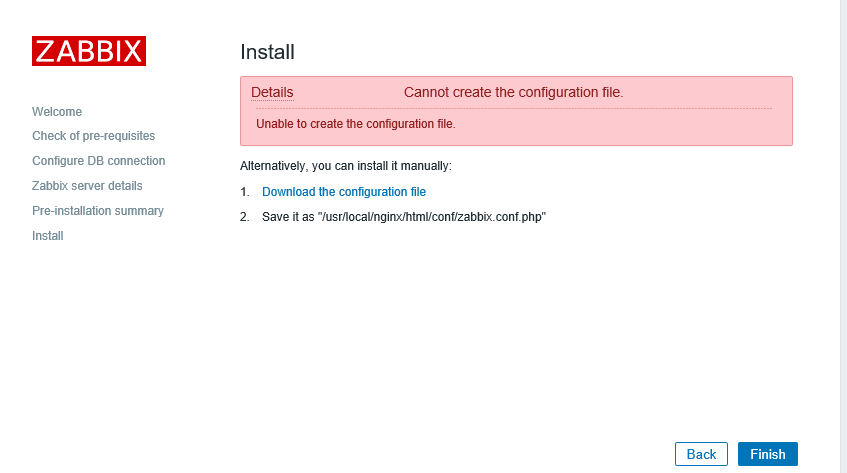


以下默认，直接下一步



下一步：





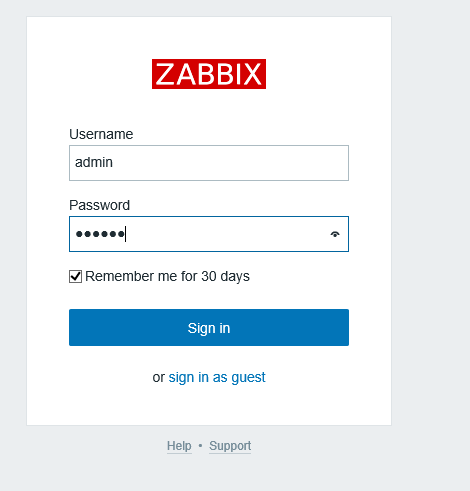
如果写入失败，直接[download](http://192.168.136.137/setup.php?save_config=1)这个配置文件，然后上传上去

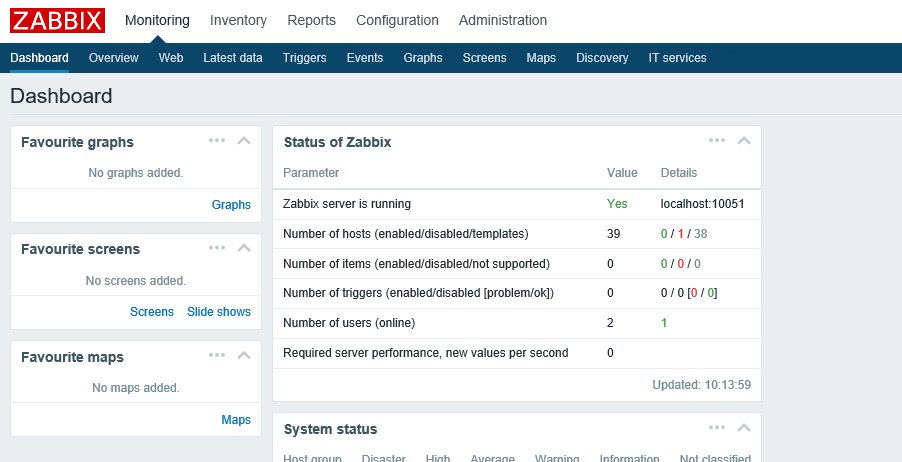
[root@xuegod63~]#mvzabbix.conf.php /usr/local/nginx/html/conf/ #然后刷新

如果显示成功

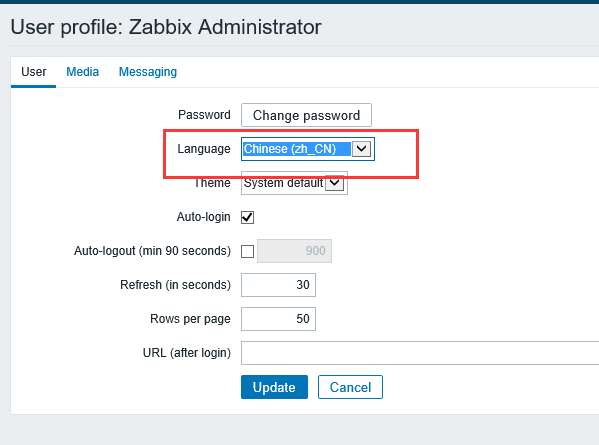
点击 Finish，完成安装！

登陆 Zabbix 并配置 登陆之后我们修改语言 默认用户名和密码分别为 admin，zabbix





点击右上角用户，然后选择语言



然后 Update 即可，页面变为中文



点击配置-主机 启用本主机（默认没有启动，点击后面红色的停用，则启用）



把zabbix\_agent 也启动起来：

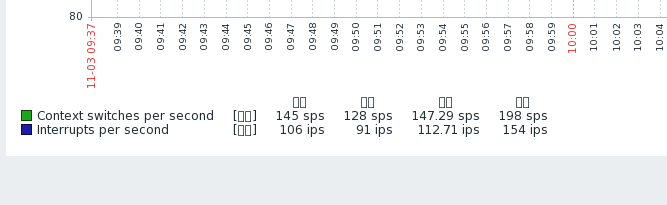
[root@xuegod63 ~]# /usr/local/zabbix/sbin/zabbix\_agentd 或者/usr/local/zabbix/sbin/zabbix\_agentd start

再次刷新页面，可以看到，agent状态已经正常



可以看到已经开始出图：





但是发现下面显示有问题:这个是中文字符集的问题，在英文状态下显示是正常的

图表字符乱码的解决办法

在 Windows 下面找一个 ttf 的字体文件C:\Windows\Fonts

[root@xuegod63 ~]# cd /usr/local/nginx/html/fonts/

[root@xuegod63 fonts]# mv DejaVuSans.ttf DejaVuSans.bak

[root@xuegod63 fonts]# mv 微软雅黑.ttf DejaVuSans.ttf

这样字符问题即解决

Zabbix 启动服务总结

Nginx 服务启动

Php-fpm 服务启动

Mysql 服务器启动

测试LNMP环境正常

导入数据库、授权、源码安装zabbix

Zabbix\_server 服务启动

配置监控远程主机：

安装 Agent

[root@xuegod69 ~]# yum install -y curl curl-devel net-snmp net-snmp-devel perl-DBI // 解决 依赖

[root@xuegod69 ~]# useradd-M -s /sbin/nologin zabbix

[root@xuegod69 ~]# tar xvf zabbix-3.4.3.tar.gz -C /usr/local/src/

[root@xuegod69 ~]# cd /usr/local/src/ zabbix-3.4.3/

[root@xuegod69 zabbix-3.4.3]#

./configure --prefix=/usr/local/zabbix --enable-agent

//被监控端只需要部署 agnet

[root@xuegod69 zabbix-3.4.3]# make install ; cd

[root@xuegod64 ~]# cat /etc/services|grep zabbix

zabbix-agent 10050/tcp # Zabbix Agent

zabbix-agent 10050/udp # Zabbix Agent

zabbix-trapper 10051/tcp # Zabbix Trapper

zabbix-trapper 10051/udp # Zabbix Trapper

启动agent：

[root@xuegod64 ~]# /usr/local/zabbix/sbin/zabbix\_agentd

也可以使用系统启动脚本来启动：

配置 Agnet 开机启动

[root@xuegod69 ~]# cp /usr/local/src/zabbix-3.4.3/misc/init.d/fedora/core5/zabbix\_agentd /etc/init.d/

[root@xuegod69 ~]# vim /etc/init.d/zabbix\_agentd

#上一步生成了启动脚本，然后现在编辑脚本

ZABBIX\_BIN="/usr/local/zabbix/sbin/zabbix\_agentd" #修改此行

[root@xuegod69 ~]# chkconfig --add zabbix\_agentd [root@xuegod69 ~]# chkconfig zabbix\_agentd on

拓展：core 和 core5 里面的文件都可以实现 zabbix 代理启动，core 里面是测试版本，core5 是稳定版本

配置 Agentd 的配置文件

[root@xuegod69~]#vim /usr/local/zabbix/etc/zabbix\_agentd.conf #客户端操作

Server=192.168.142.125 #zabbix server 地址，可以多个，用，隔开

ServerActive=192.168.142.125 #主动检查的意思,主动检查主机的数据的数据发送给 Zabbix Server

Hostname=xuegod69

UnsafeUserParameters=1

启动服务

[root@xuegod69 ~]#/etc/init.d/zabbix\_agentd restart [root@xuegod69 ~]# netstat -antup | grep zabbix

tcp 0 0 0.0.0.0:10050 0.0.0.0:\* LISTEN

9683/zabbix\_agentd

Server 端测试通讯

[root@xuegod63 ~]# /usr/local/zabbix/bin/zabbix\_get -s 192.168.0.64 -p10050 -k system.uname

#可以获取对方系统版本则证明没有问题

服务端配置管理脚本

[root@xuegod63 ~]# cp/usr/local/src/zabbix-3.4.3/misc/init.d/fedora/core/zabbix\_agentd/etc/init.d/

[root@xuegod63 ~]# vim /etc/init.d/zabbix\_agentd

BASEDIR=/usr/local/zabbix #修改此行内容为 zabbix 的安装位置

[root@xuegod63 ~]# chkconfig --add zabbix\_agentd [root@xuegod63 ~]# chkconfig zabbix\_agentd on

[root@xuegod63 ~]# /etc/init.d/zabbix\_agentd restart //可以监控自己，也要启动服务

[root@xuegod63 ~]# netstat -antup | grep zabbix //两者之间监控端口不一样

tcp 0 0 0.0.0.0:10050 0.0.0.0:\* LISTEN

51645/zabbix\_agentd

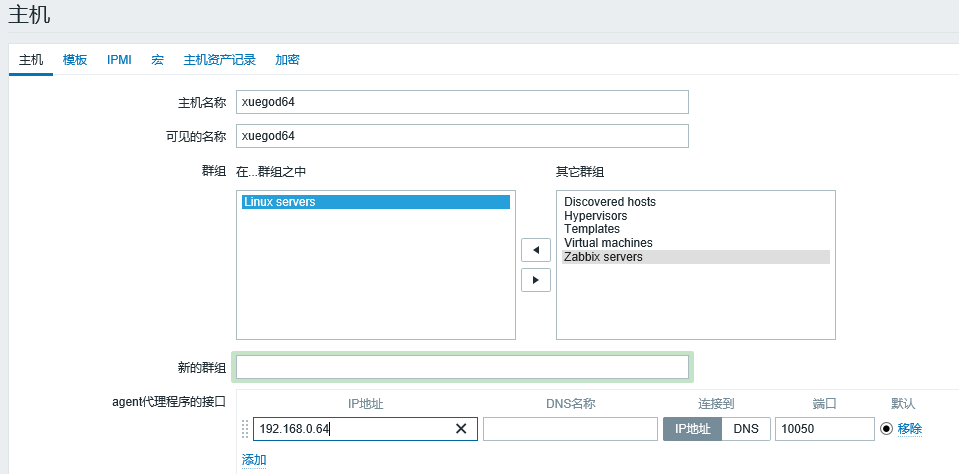
tcp 0 0 0.0.0.0:10051 0.0.0.0:\* LISTEN

10786/zabbix\_server

监控自己使用 10050 端口，监控对方使用的是 10051 端口

添加 Host 主机

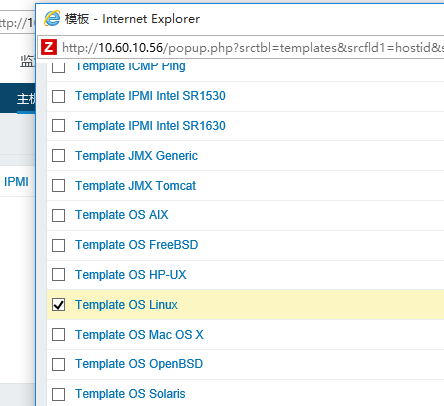


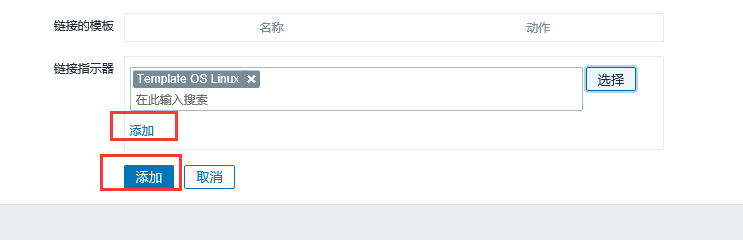


选择模块：



在弹出的模块里选择一个自己需要的模块：





返回之后，可以看到已经添加进来：等到可用性变成绿色即可出图监控





这时间可以通过图形功能来查看出图情况：



=========通过以上步骤成功添加一台远程监控主机======

添加监控项：

通过比较zabbix servers 群组和Linux servers 群组，可以发现，默认的配置中Linux servers监控项比较少，下面手动添加监控项：

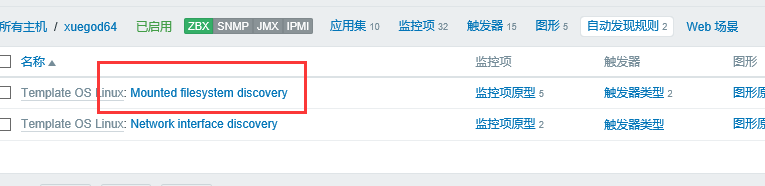


筛选无触发器，出来很多监控项，默认都是启用状态，我们无需再次启动，下面修改自动发现的时间





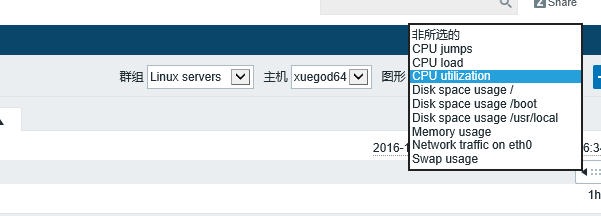
文件系统探测和网络流量探测



把原来的3600s改成10s



修改更新数据后，再次返回图形监控页面，稍等几分钟再次查看监控项



再次返回主页，可以看到整体的监控状态

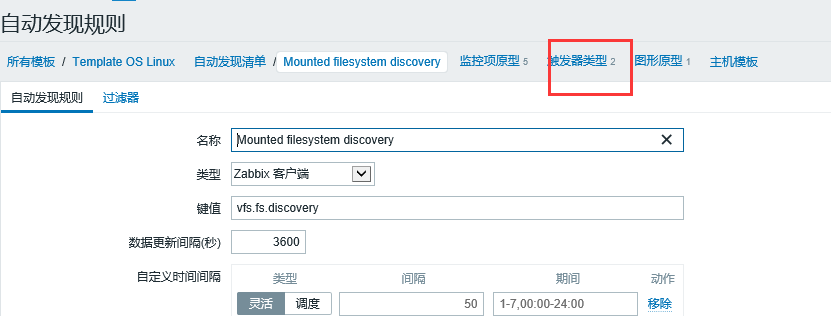


设置 Triger 触发器值



找到当前主机应用的模板：





默认的值是20%



修改成自己设定的值



更新数据

========以上是触发器值的设置================

自定义绘图颜色：

找到主机对应的模块---修改模板对应的图形颜色





根据需求修改每一个项的显示颜色



总结 zabbix 的文件

zabbix\_agent.conf 是用超级服务（xinetd）的方式来启动的，被动检查，只有 Server 说检查的时候才检查

zabbix\_agentd.conf 是以独立进程的方式来启动的，一般使用这个来做配置，自动检查，自动提交 zabbix\_server.conf Server 启动配置文件，只有一个

zabbix\_get 是 Zabbix 中的一个程序，用于 Zabbix-Server 到 Zabbix-Agent 的数据获取，通常可以用来 检测验证 Agent 的配置是否正确。它的使用方法如下

zabbix\_get [-hV] -s <host name or IP> [-p <port>] [-I <IP address>] -k <key>

-h：远程 Zabbix-Agent 的 IP 地址或者是主机名。

-p：远程 Zabbix-Agent 的端口。

-I：本机出去的 IP 地址，用于一台机器中有多个网卡的情况。

-k：获取远程 Zabbix-Agent 数据所使用的 Key。

[expmple@~]# zabbix\_get -s 192.168.0.64 -k system.uname

zabbix\_sender 与 get 相反，用来发送 Zabbix 服务器处理性能数据。该工具通常用于长时间运行的用户脚 本，用于定期发送可用性和性能数据。使用方法如下

zabbix\_sender [-Vhv] {[-zpsI] -ko | [-zpI] -T -i <file> -r} [-c <file>]

参数说明：

-c --config <file> 配置文件绝对路径

-z --zabbix-server <server> zabbix server 的 IP 地址

-p --port <server port> zabbix server 端口.默认 10051

-s --host <hostname> 主机名，zabbix 里面配置的主机名（不是服务器的 hostname），不能使用

ip 地址

-I --source-address <IP address> 源 IP

-k --key <key> 监控项的 key

-o --value <key value> key 值

-i --input-file <input file> 从文件里面读取 hostname、key、value 一行为一条数据，使用空格作为分

隔符，如果主机名带空格，那么请使用双引号包起来

-T --with-timestamps 一行一条数据，空格作为分隔符: <hostname><key><timestamp>

<value>，配合 --input-file option，timestamp 为 unix 时间戳

-r --real-time 将数据实时提交给服务器

-v --verbose 详细模式, -vv 更详细

配置 Zabbix 监控 Apache 服务

http://www.zabbix.org/wiki/Main\_Page

<https://github.com/zabbix/zabbix-community-repos>

各种模板

<http://www.douglas.wiki.br/doku.php?id=en:installing_and_configuring_zabbix>

下载脚本wget https://github.com/lorf/zapache/archive/master.zip

启用apache 服务器状态，开启apache的server-status

[root@xuegod69 httpd]# vim conf/httpd.conf #在末尾加入

ExtendedStatus On

<location /server-status>

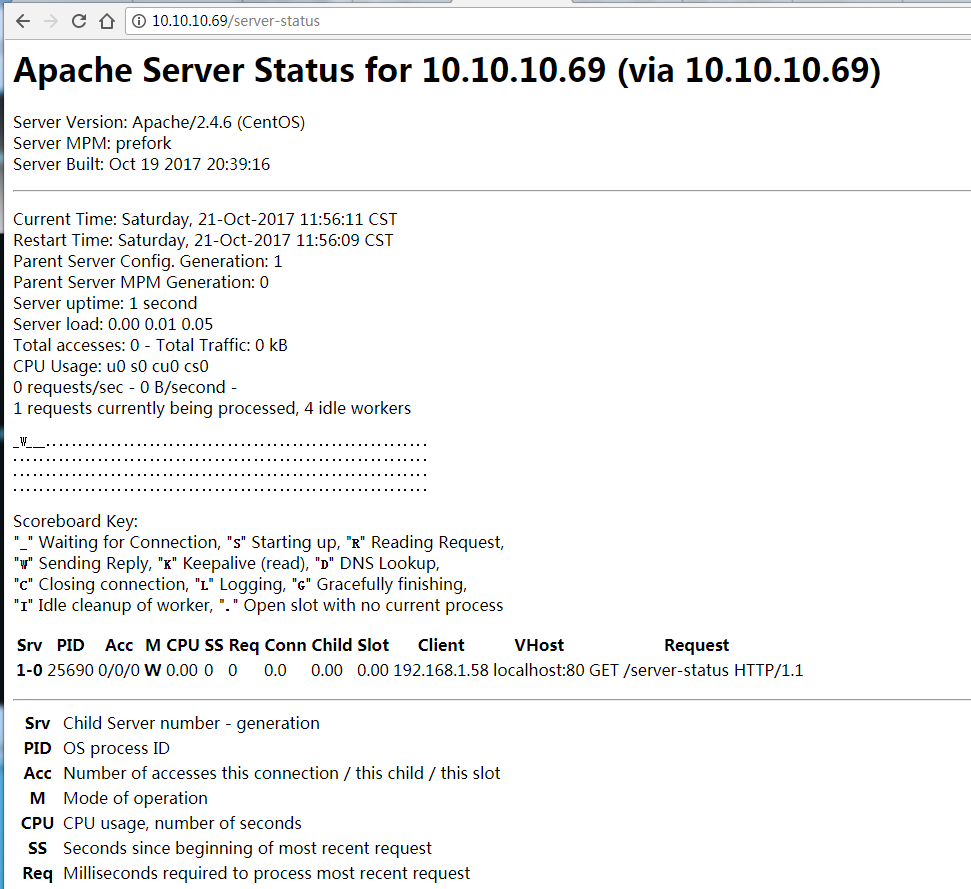
SetHandler server-status

Order allow,deny

Allow from 127.0.0.1 192.168.1.0/24

</location>

[root@xuegod69 bin]# systemctl restart httpd 重启后测试



[root@xuegod69 ~]# unzip master.zip #解压下载的包

[root@xuegod69 ~]# cd zapache-master/

[root@xuegod69 zapache-master]# ls

httpd-server-status.conf.sample zapache

README.md zapache-template-active.xml

userparameter\_zapache.conf.sample zapache-template.xml

说明

httpd-server-status.conf.sample 是用于配置server-status的上个步骤已经配置过了

userparameter\_zapache.conf.sample 、zapache 关键的文件  
zapache-template-active.xml  zapache-template.xml模板

**将解压出来的几个文件放到相应目录**

[root@xuegod69 zapache-master]#cp zapache /usr/local/bin/

[root@xuegod69 zapache-master]# cp userparameter\_zapache.conf.sample /usr/local/zabbix/etc/zabbix\_agentd.conf.d/

[root@xuegod69 zapache-master]# cd !$

[root@xuegod69 zabbix\_agentd.conf.d]# mv userparameter\_zapache.conf.sample userparameter\_zapache.conf  重命名

[root@xuegod69 zabbix\_agentd.conf.d]# vim userparameter\_zapache.conf

UserParameter=zapache[\*],/var/lib/zabbixsrv/externalscripts/zapache \$1

改为UserParameter=zapache[\*],/usr/local/bin/zapache \$1

查看是否有执行权限

没有权限可执行下面命令

[root@xuegod69 ~]# chmod +x /usr/local/bin/zapache

修改zabbix -agentd.conf

[root@xuegod69 ~]# vim /usr/local/zabbix/etc/zabbix\_agentd.conf

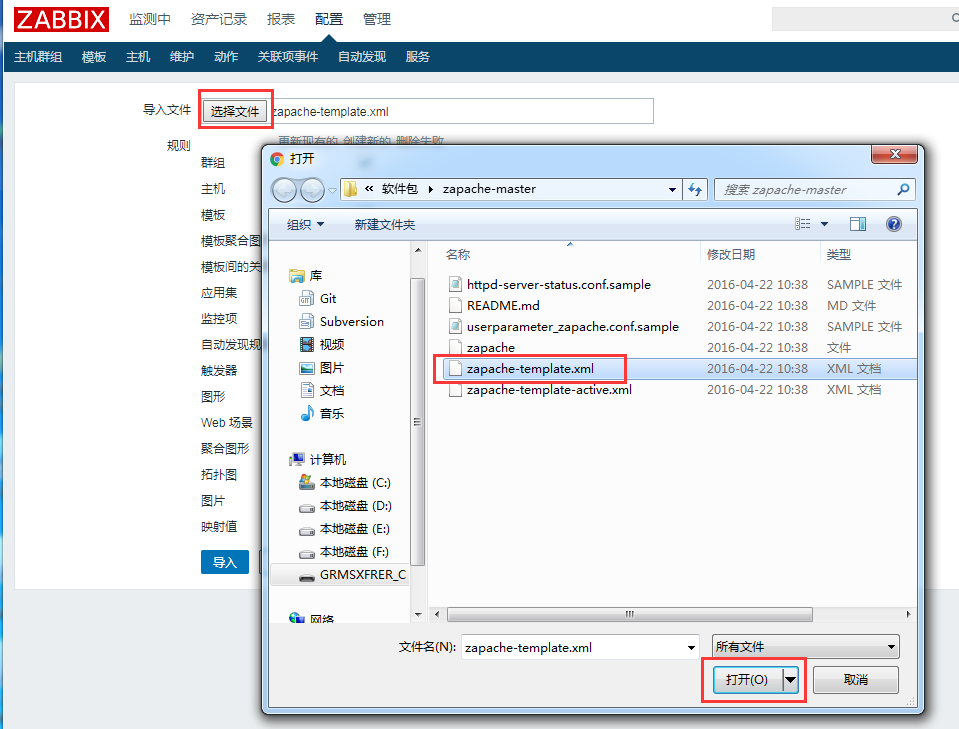
Include=/usr/local/zabbix/etc/zabbix\_agentd.conf.d/

#启用引用此目录文件

前端页面操作

导入模板

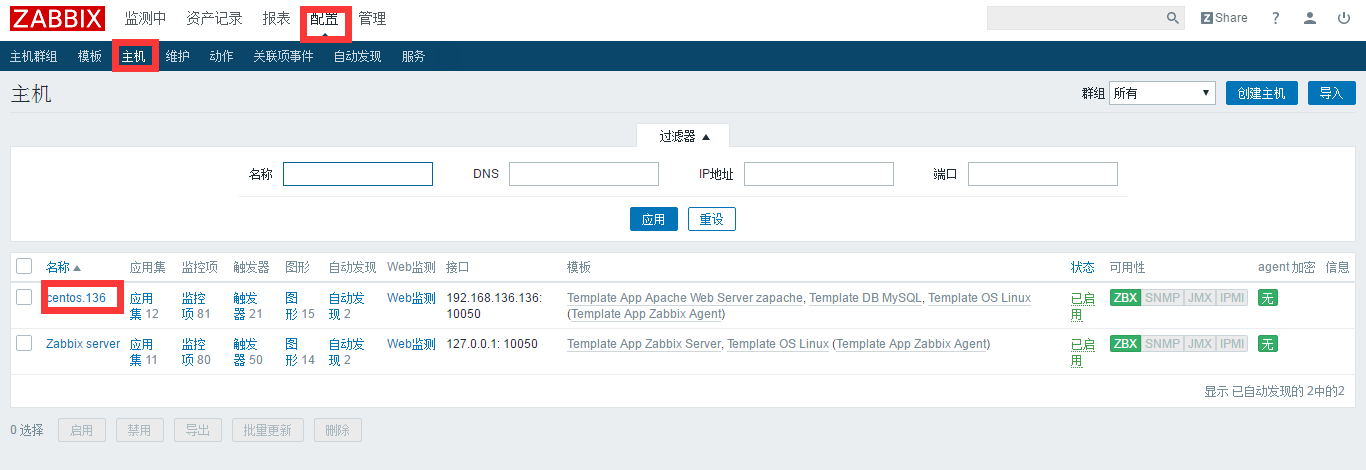




关联模板

配置-主机（选择apache客户端）

模板-选择（选择上面导入的模板）-添加-更新

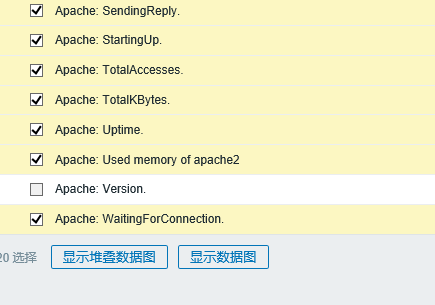
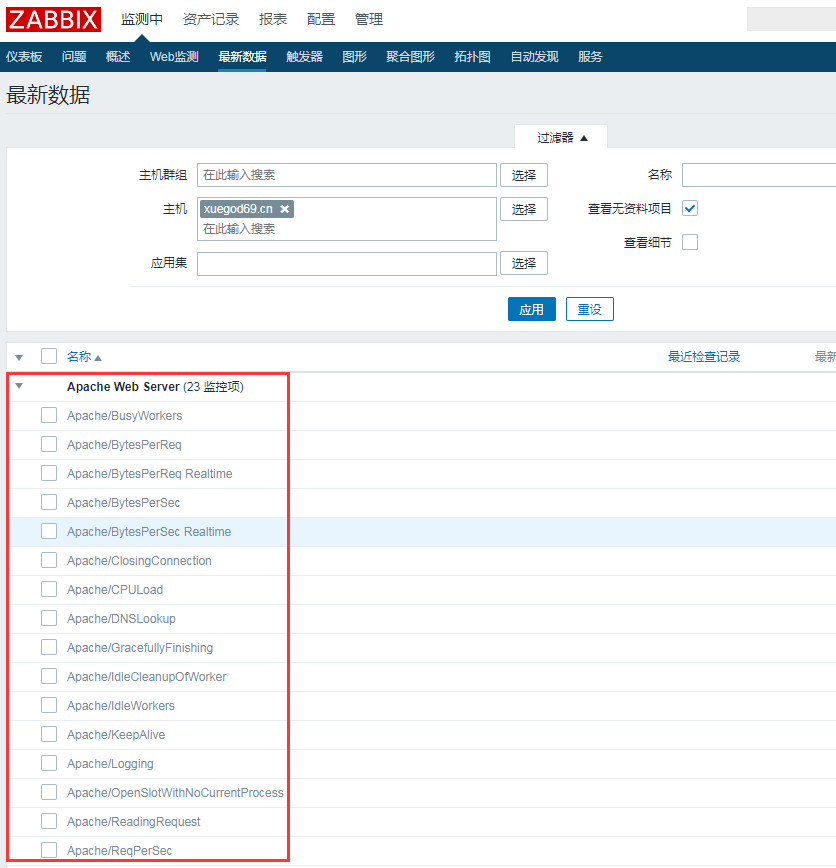




查看最新数据，筛选Apache



可以选择生产图形



配置 zabbix 监控 MySQL

和监控我们的 Apache 一样，寻找合适的脚本，不过，脚本并不是那么容易就可以找得到的! 要么自己手动去写，要么自己找到的拿来修改，都是需要花费代价的！ZABBIX 默认提供了 MYSQL 的监控模板

数据库用户授权

mysql> grant usage on \*.\* to zabbix@'%' identified by '123456';

mysql> flush privileges;

[root@xuegod69 ~]#cp /usr/local/src/zabbix-3.4.3/conf/zabbix\_agentd/userparameter\_mysql.conf /usr/local/zabbix/etc/zabbix\_agentd.conf.d/

[root@xuegod69~]#vim /usr/local/zabbix/etc/zabbix\_agentd.conf

Include=/usr/local/zabbix/etc/zabbix\_agentd.conf.d/\*.conf #启用并修改

建立数据库和 zabbix 的链接信息

[root@xuegod69 ~]# vim /usr/local/zabbix/etc/.my.cnf

#Zabbix Agent

[mysql]

host=xuegod63

user=zabbix

password=123456

socket= /var/lib/mysql/mysql.sock

[mysqladmin]

host=xuegod63

user=zabbix

password=123456

socket= /var/lib/mysql/mysql.sock

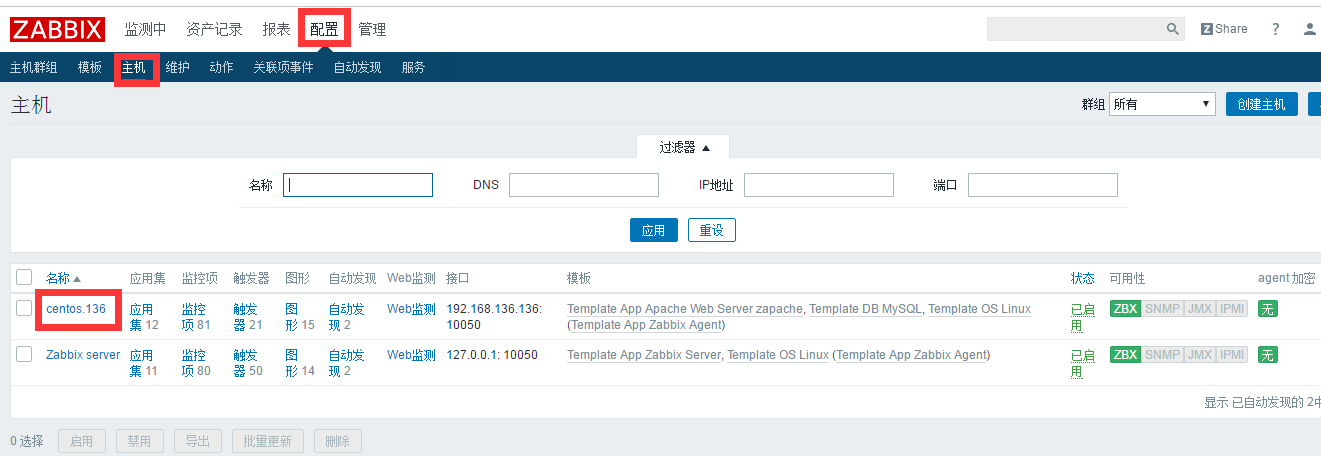
[root@xuegod69 ~]# vim /usr/local/zabbix/etc/zabbix\_agentd.conf.d/userparameter\_mysql.conf

将 HOME=/var/lib/zabbix 全部改成 HOME=/usr/local/zabbix/etc/

# For all the following commands HOME should be set to the directory that has .my.cnf file with password information.（提示目录下必须有.my.cnf）

[root@xuegod69 ~]# /etc/init.d/zabbix\_agentd restart

模板应用主机：







Zabbix邮件报警配置：

邮件服务可以使用系统自带的邮件服务来发送邮件或者使用其他邮件服务调用第三方邮件来发送警告邮件

这里演示如何调用第三方邮件服务来发送邮件：

**第一步：首先安装mailx组件并配置好能够通过三方邮箱发送邮件**

[root@xuegod63 ~]# yum install mailx

然后编辑mailx的配置文件

[root@xuegod63 ~]# vi /etc/mail.rc

在最末尾添加如下

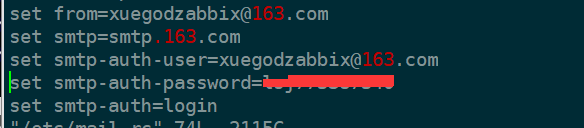
信息

set from=xuegodzabbix@163.com

set smtp=smtp.163.com

set smtp-auth-user=xuegodzabbix@163.com

set smtp-auth-password=你的密码 注意：此为163邮箱设置的客户端授权密码

set smtp-auth=login 

[**这里的xxxxxx@163.com**](mailto:这里的xxxxxx@163.com) **表示发送账号**

保存退出后测试邮件是否能够正常发送出去

[root@xuegod63 ~]# echo "zabbix test mail" |mailx -s "zabbix"XXXX@qq.com

在zabbix服务端写邮件发送脚本：

[root@xuegod63 ~]# cd /usr/local/zabbix/share/zabbix/alertscripts/

[root@xuegod63 alertscripts]# vim sendmail.sh

#!/bin/bash

messages=`echo $3 | tr '\r\n' '\n'`

subject=`echo $2 | tr '\r\n' '\n'`

echo "${messages}" | mailx -s "${subject}" $1 >>/tmp/sendmail.log 2>&1

[root@xuegod63 alertscripts]# chown zabbix.zabbix sendmail.sh

[root@xuegod63 alertscripts]# chmod +x sendmail.sh

修改主配置文件，让服务读取到这个脚本：

vim /usr/local/zabbix/etc/zabbix\_server.conf

修改：

# AlertScriptsPath=${datadir}/zabbix/alertscripts

为

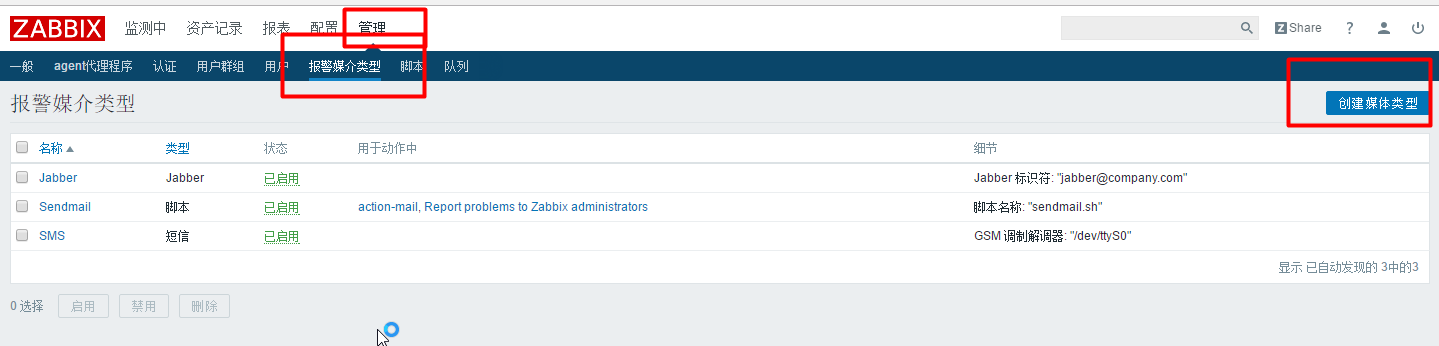
AlertScriptsPath=/usr/local/zabbix/share/zabbix/alertscripts

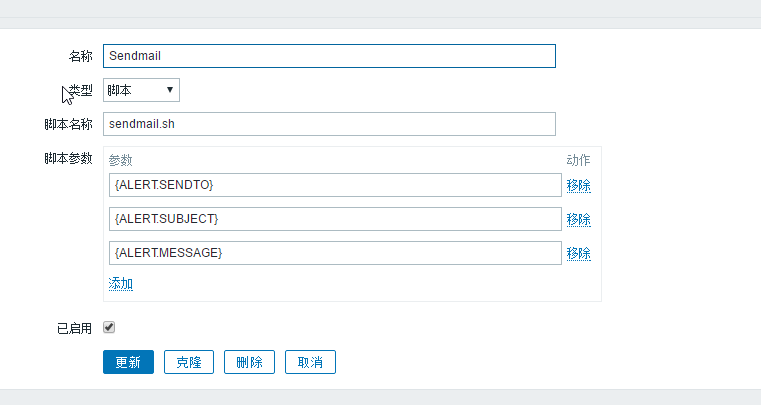
[root@xuegod63alertscripts]#chmod 777 /tmp/sendmail.log

**测试发送脚本**

/usr/local/zabbix/share/zabbix/alertscripts/sendmail.sh 136xxx@126.com "测试邮件标题" "测试邮件内容"

页面添加：





{ALERT.SENDTO}

{ALERT.SUBJECT}

{ALERT.MESSAGE}

很多人安装zabbix 3.0之后，写的脚本一直发信不成功,手动执行时可以的。

这是因为zabbix3.0之后，可以自定义参数了。所以不写参数，它是不会传参数的。

在2.x版本不存在这个问题，默认会传3个参数。



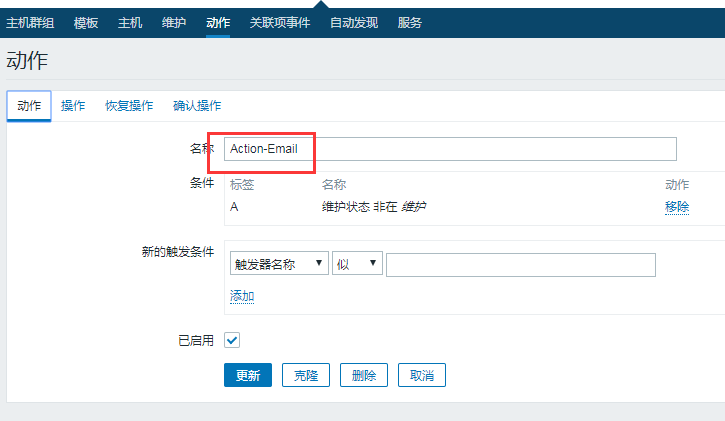




内容如下

动作

名称：Action-Email



操作

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}

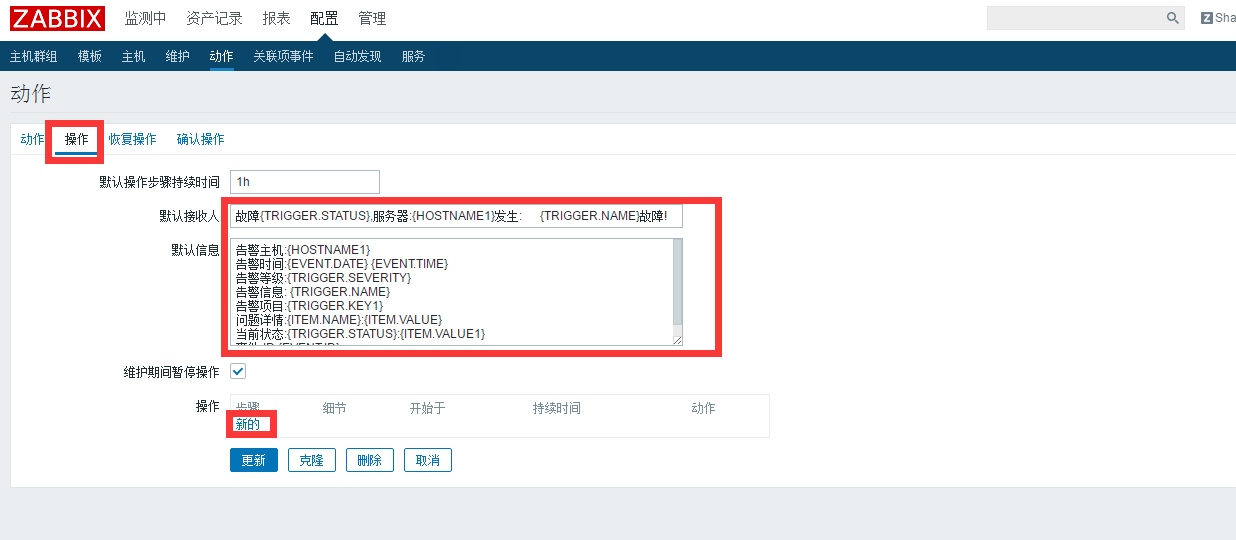
告警信息: {TRIGGER.NAME}

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}



恢复操作

默认接收人：已恢复！{TRIGGER.STATUS}, 服务器:{HOSTNAME1}: {TRIGGER.NAME}

恢复信息：

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}

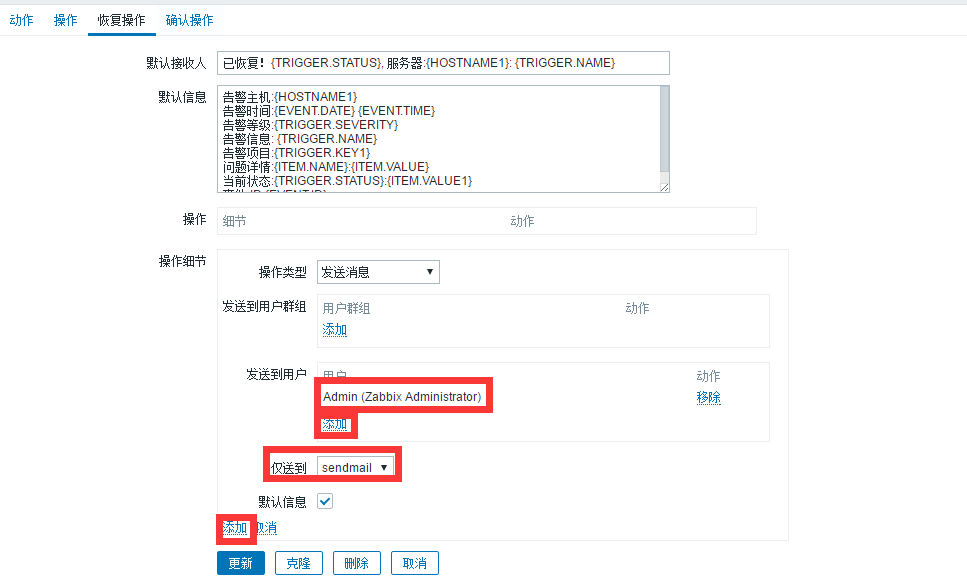
告警项目:{TRIGGER.KEY1}

问题详情:{ITEM.NAME}:{ITEM.VALUE}

当前状态:{TRIGGER.STATUS}:{ITEM.VALUE1}

事件 ID:{EVENT.ID}







测试，关闭 zabbix 客户端服务 zabbix\_agentd

[root@xuegod69 ~]# /etc/init.d/zabbix\_agentd stop 结果：

延迟比较大，大概要5-10分钟

可以在日志看送达状态：



参考：

<http://thedream.blog.51cto.com/6427769/1827984>

<http://xiao987334176.blog.51cto.com/2202382/1768339>

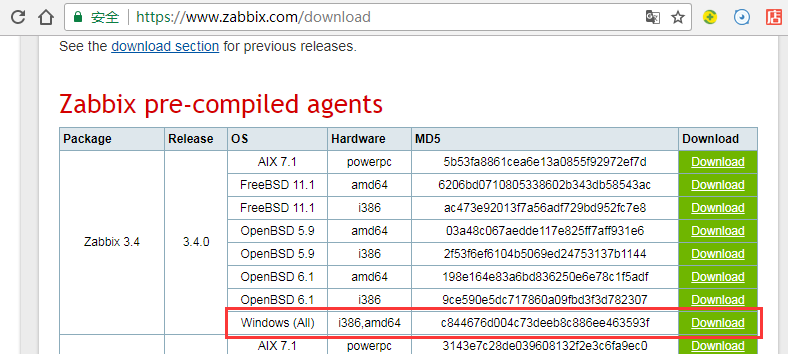
<http://clovemfong.blog.51cto.com/3297559/1702105>

<https://ywwd.net/read-1044>

<http://www.cnblogs.com/Eivll0m/p/5446993.html>

配置 zabbix 自动发现并监控（监控Ｗｉｎｄｏｗｓ为案例）

首先下载安装Ｗｉｎｄｏｗｓ端ａｇｅｎｔ



找到conf下的配置文件 zabbix\_agentd.win.conf (建议用文本编辑软件打开)，修改LogFile、Server、ServerActive、Hostname这四个参数。具体配置如下：  
LogFile=c:\zabbix\_agentd.log  
Server=192.168.142.125  
Hostname=download  
ServerActive=192.168.142.125#zabbix server地址  
其中logfile是zabbix日志存放地址。Server 是zabbix服务端ip地址。Hostname是本机机器名。

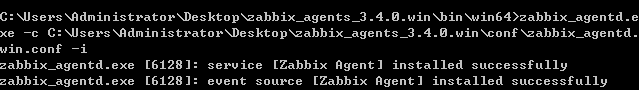
一定要以管理员身份来运行CMD，否则安装不了

cd C:\Users\Administrator\Desktop\zabbix\_agents\_3.4.0.win\bin\win64



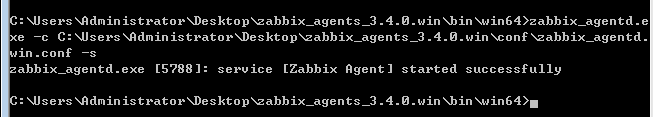
安装windos zabbix客户端

zabbix\_agentd.exe –c C:\Users\Administrator\Desktop\zabbix\_agents\_3.4.0.win\conf\zabbix\_agentd.win.conf–i

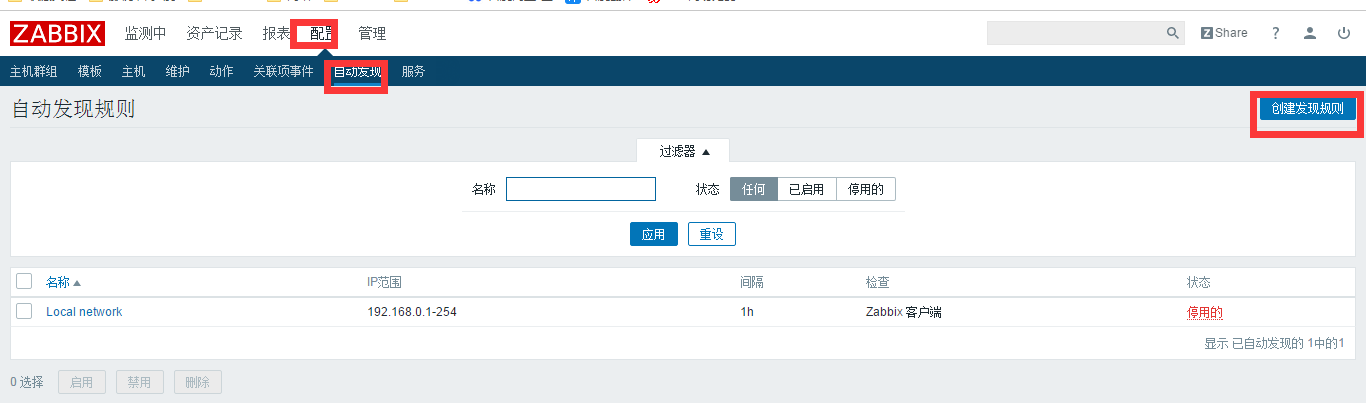


启动客户端

zabbix\_agentd.exe –c C:\Users\Administrator\Desktop\zabbix\_agents\_3.4.0.win\conf\zabbix\_agentd.win.conf–s



创建自动发现规则





添加关联动作





操作-可添加发现主机发送邮件通知

发送模版

默认接收人自动发现主机: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}

默认信息

发现规则: {DISCOVERY.RULE.NAME}

设备IP:{DISCOVERY.DEVICE.IPADDRESS}

设备DNS: {DISCOVERY.DEVICE.DNS}

设备状态: {DISCOVERY.DEVICE.STATUS}

设备运行时间: {DISCOVERY.DEVICE.UPTIME}

设备服务端口: {DISCOVERY.SERVICE.NAME}



等待一会，就可以发现自动探测监控了

推荐一款监控：<http://www.weadmin.com/>