**Morning：**

1、实现Zabbix报警功能

**-报警的条件**

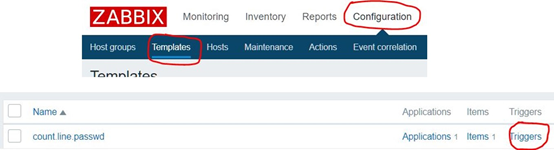
触发器+报警动作。

触发器：表达式。当触发条件发生后，会导致一个触发事件；

动作：触发器的条件被触发后的行为(发邮件、重启服务等)

**步骤1：创建触发器规则**

1. 创建触发器



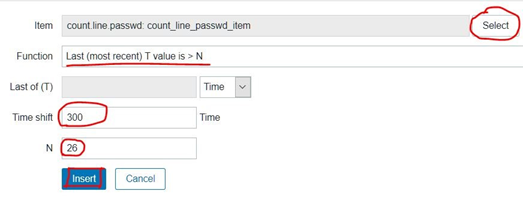
2）触发器表达式

{<server>:<key>.<function>(<parameter>)}<operator><constant>

#{主机：key.函数(参数)}<表达式>常数

3）配置触发器

填写Name --> 选择saveity(严重性) --> 添加表达式



#监控账户数量在最近300s内大于26报警

#配置触发器时最好使用英文页面

**步骤2：设置邮件**

1）设置邮件服务器



#配置邮件服务器、发件人账户(可以是外部邮箱，设置正确的端口)

2）添加收件用户



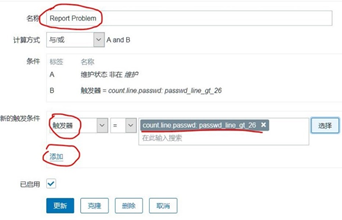


**步骤3：创建Action动作**

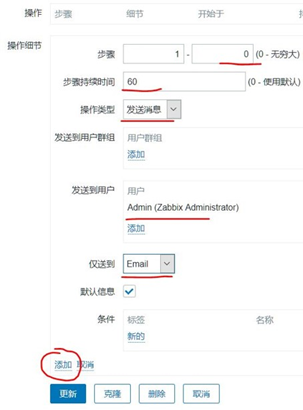
1）创建Action动作



2）配置触发条件



3）配置Action动作的具体操作



#步骤：发送多少次邮件(1-0为无限次)

4）测试效果



#在Zabbix\_Server查看报警邮件

2、Zabbix自动发现

**步骤1：自动发现规则**

1）创建自动发现规则



2）填写规则

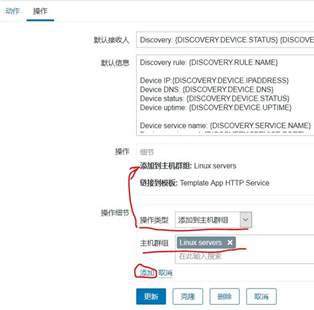
#检查哪个网段，检查类型是什么(icmp、http等)

**步骤2：创建动作**

1）创建自动发现动作



2）配置Action具体行为

3、Zabbix主动监控

**-主动与被动监控**

被动监控：Server向Agent发起连接，发送监控key，Agent接受请求，响应监控数据。（默认）

主动监控：Agent向Server发起连接，Agent请求监控项目列表，Server响应Agent发送一个items列表，Agent确认收到监控列表，TCP连接完成，会话关闭，Agent开始周期性地收集数据。

区别：Server不用每次需要数据都连接Agent，Agent会自己收集数据并处理数据，Server仅需要保存数据即可。

当监控主机达到一定量级后，Zabbix服务器会越来越慢，此时，可以考虑使用主动监控，释放服务器的压力。

**步骤1：添加被监控主机**

1）部署Zabbix\_agent

参考：阶段3 --> 4.SECURITY --> DAY05 --> Page04

2）修改agent配置文件

[root@zab\_cli2 ~]# vim /usr/local/etc/zabbix\_agentd.conf

#Server=127.0.0.1,192.168.2.5 #注释掉，不允许被监控

StartAgents=0 #定义被动监控时启动的zabbix\_agentd进程数为0

ServerActive=192.168.2.5 #允许哪些主机监控本机(主动模式)

Hostname=Zabbix\_cli2 #和zabbix服务器配置的主机名称一致

RefreshActiveChecks=120 #默认120s检测一次

UnsafeUserParameters=1 #允许自定义key

Include=/usr/local/etc/zabbix\_agentd.conf.d/ #自定义key目录

[root@zab\_cli2 ~]# killall zabbix\_agentd

[root@zab\_cli2 ~]# zabbix\_agentd

#重启服务后，zabbix\_agentd进程数为0

**步骤2：创建主动监控的模板**

1）克隆系统模板

配置 --> 模板 -->选择Template OS Linux --> 全克隆

新模板名称为：Template OS Linux Active --> 添加

2）修改新模板监控项模式

配置 --> 模板 --> 选择新克隆的模板

Items（监控项）--> 全选 --> 批量更新 --> 类型改为主动式



3）禁用被动监控项目



#某些监控项不支持主动模式，直接停用

**步骤3：添加监控主机**

#主机名称与配置文件Hostname一致

#agent代理程序接口：IP地址0.0.0.0，端口0

#选择新模板Template OS Linux Active

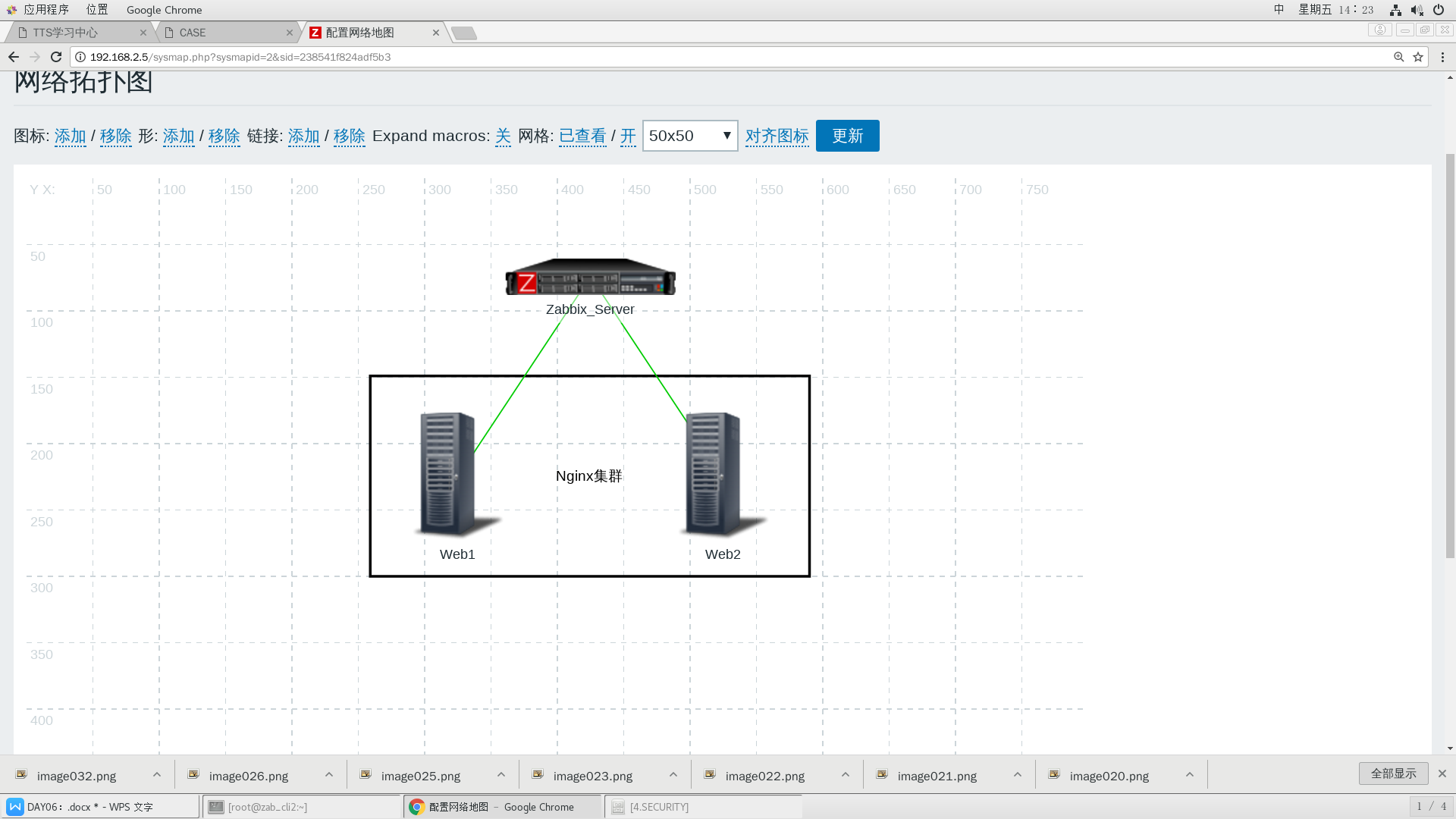
参考：阶段3 --> 4.SECURITY --> DAY05 --> Page5~6

**Afternoon：**

1、拓扑图与聚合图形

**步骤1：创建拓扑图**

image032



画图工具：

Icon（图标），添加新的设备后可以点击图标修改属性

Shape（形状）

Link（连线），先选择两个图标，再选择连线

**步骤2：创建聚合图形**



Owner：设置为默认的Admin用户

Name：设置为zabbixclient\_web2\_host

Columns：设置为2列

Rows：设置为2行



2、自定义监控案例

**步骤1：监控Nginx服务状态**

1）Nginx安装status模块

参考：阶段2 --> 2.OPERATION --> DAY03 --> Page04优化2

2）自定义监控key

语法格式：**UserParameter=key[\*],<command>**

#key中的参数，会传递到后面命令的位置变量

[root@zab\_cli1 ~]# vim /usr/local/etc/zabb...conf.d/naginx.status

UserParameter=nginx.status[\*],/tmp/nginx\_status.sh $1

[root@zab\_cli1 ~]# /etc/init.d/zabbix\_agentd restart

#指定自定义key名称与脚本，重启服务

[root@zab\_cli1 ~]# vim /tmp/nginx\_status.sh

#!/bin/bash

case $1 in

active)

curl -s http://192.168.2.100/status | awk 'NR==1{print $NF}';;

waiting)

curl -s http://192.168.2.100/status | awk 'NR==4{print $NF}';;

accepts)

curl -s http://192.168.2.100/status | awk 'NR==3{print $1}'

esac

[root@zab\_cli1 ~]# chmod +x /tmp/nginx\_status.sh

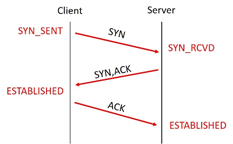
#自定义监控脚本

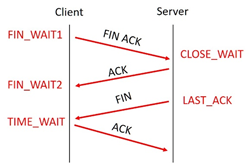
[root@zab\_cli1 ~]# zabbix\_get -s 127.0.0.1 -k 'nginx.status[accepts]'

#测试效果

**步骤2：监控网络连接状态**

1）TCP的3次握手、4次断开





2）查看网络连接状态

[root@zab\_cli1 ~]# ss -anpt

#参考阶段2 --> 2.OPERATION --> DAY01 --> netstat

#不加-l可以显示所有连接状态

[root@zab\_cli1 ~]# ab -c 1000 -n 10000 http://192.168.2.100/

#模拟多人并发连接

3）自定义监控脚本

[root@zab\_cli1 ~]# vim /tmp/net\_status.sh

#!/bin/bash

case $1 in

estab)

ss -antp | awk '/^ESTAB/{x++} END{print x}';;

time\_wait)

ss -antp | grep "^TIME-W" | wc -l;;

esac

[root@zab\_cli1 ~]# chmod +x /tmp/net\_status.sh

[root@zab\_cli1 ~]# /usr/local/etc/zabbix\_agentd.conf.d/net.status

UserParameter=net.status[\*],/tmp/net\_status.sh $1

[root@zab\_cli1 ~]# /etc/init.d/zabbix\_agentd restart

4）监控net.status

Configuration --> Hosts点击主机后面的items --> 点击Create item

