**Zabbix 自动发现和action**

1：自动发现规则的创建(网络发现)





参数解析：

Name：规则的惟一名称。例如“Local network”

Discovery by proxy：通过什么来执行发现，是没有代理还是通过代理proxy来执行发现

IP range：用于发现的IP地址范围。 它可能有以下格式：单IP：192.168.1.33、

IP地址范围：192.168.1-10.1-255。、IP掩码：192.168.4.0/24、列表：192.168.1.1-255,192.168.2.1-100,192.168.2.200,192.168.4.0/24

Delay (in sec)：此参数定义Zabbix执行规则的频率。在先前发现实例的执行结束之后测量延迟，因此没有重叠。默认是3600秒即一小时发现一次。

Checks：Zabbix将使用此列表进行发现。

支持的检查：SSH，LDAP，SMTP，FTP，HTTP，HTTPS，POP，NNTP，IMAP，TCP，Telnet，Zabbix代理，SNMPv1代理，SNMPv2代理，SNMPv3代理，ICMP ping。

Device uniqueness criteria：设备唯一的名称。IP地址 - 不处理多个单IP设备。 如果具有相同IP的设备已经存在，则会被认为已经被发现，并且不会添加新的主机。发现类型检查 - SNMP或Zabbix代理检查。

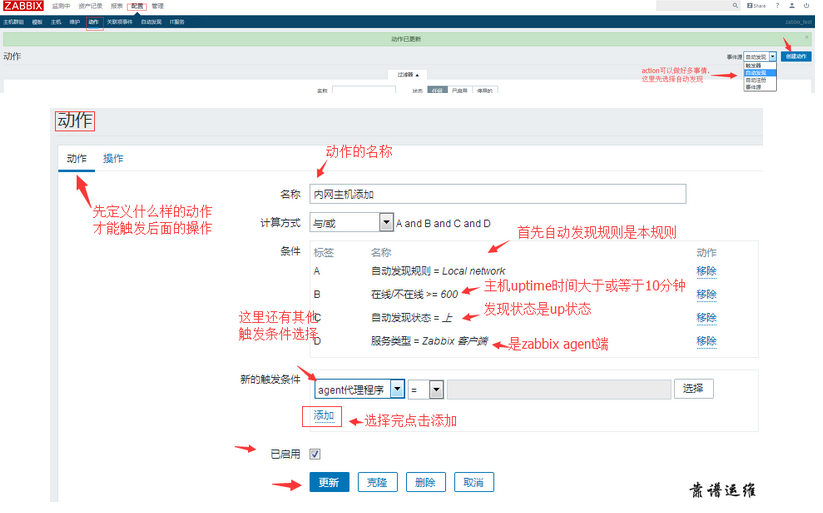
Enabled：发现规则是否是启用状态。

2：客户端的配置

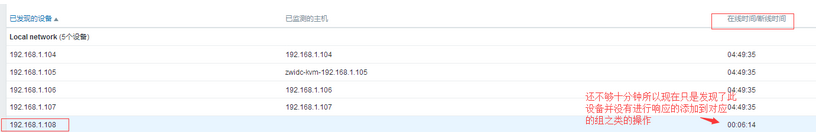
Server=192.168.1.103 #在Server这里指定下服务器的IP地址

****

3：创建动作

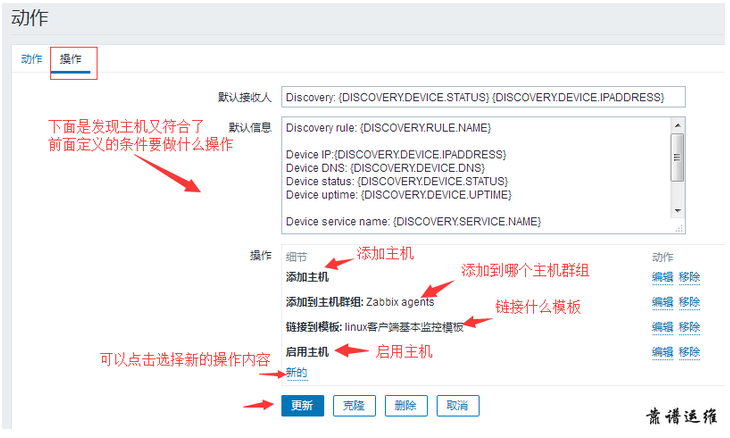


#注意上面哪个uptime时间并不是机器本身运行存活的时间，而是被发现存活的时间，如我们新发现了一台机器并不立马执行操作，要等到其值为10分钟的时候才会操作，如下图：



#从上图可以看到只是发现了设备，已检测的主机那里还是空的，说明还没有把这个主机添加到监控里面，要等到它符合了条件才会添加到监控里面去。

下面是条件已经满足之后要执行的操作：



最后看一下自动发现后添加主机的效果：



主动注册

主动注册（agent auto-registration）功能主要用于Agent主动且自动向Server注册。与前面的Network discovery具有同样的功能，但是这个功能更适用于特定的环境，当一个条件未知（如agent端的IP地址段、agentg端的操作系统版本等信息）时，Agent去请求Server仍然可以实现自动添加监控的功能。比如云环境下的监控，云环境中，IP分配就是随机的，这个功能就可以很好的解决类似的问题，如果云主机要加到zabbix监控里面来的话。

使用Host metadata注册形式

Host metadata是zabbix 2.2新增加的功能，该功能在zabbix-agent端可以自定义条件，在选择自动注册的时候，zabbix-server端可以根据Host metadata来选择条件，从而实现更多的条件筛选。

1：客户端的配置

Server=192.168.1.103

ListenPort=10050

ListenIP=192.168.1.102

ServerActive=192.168.1.103 #这里主动模式下zabbix服务器的地址

Hostname=zwidc\_web\_192.168.1.102.test

HostMetadata=linux zabbixtest.youshi #这里其实就是类似于snmp的那个public口令一样，这里设置了两个元数据，一个是告诉自己是linux服务器另一个就是写一个通用的带有公司标识的字符串

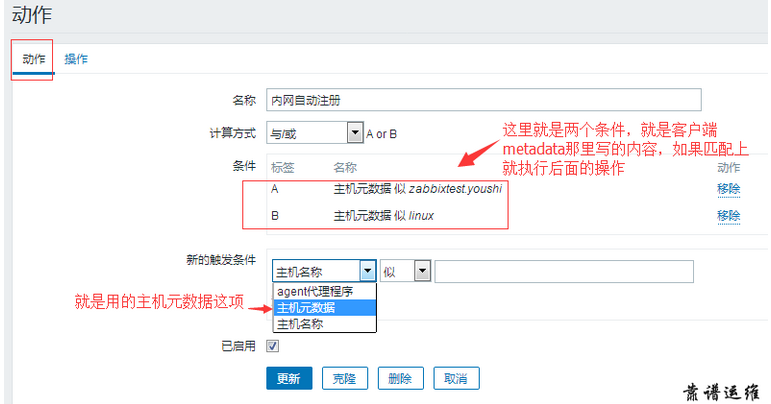
#自动注册和自动发现对于自动创建主机的时候，主机名的区别：

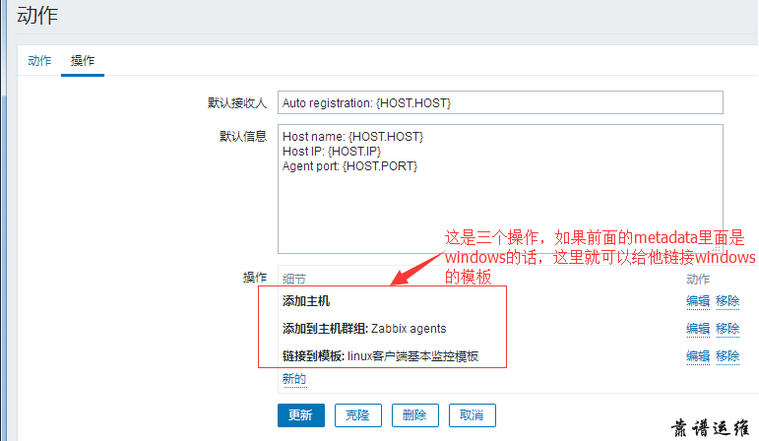
#如上面的Hostname=zwidc\_web\_192.168.1.102.test，我们想自动注册的动作所创建的主机名的名称是zwidc\_web\_192.168.1.102.test而非192.168.1.102，因为我们最终集群是要变成客户端主动发送数据给proxy端，然后proxy端主动提交数据到服务端，所以如果我们想让主机自动创建的时候是这种带有机房标识的方式，就要选择自动注册的形式，自动发现zabbix服务端自动添加主机的名称是IP，因为主动默认主机名不对称就会提交不了数据了。

#然后再自动注册动作那里除了选择元数据以外，还要选择proxy代理，这样如果是多机房的话，在添加主机的时候就会给它分配好对应机房的proxy代理。

2：服务端的操作







最终效果

