

# Zabbix

Encore à faire:

- Icones ?
- Francisation?.
- Surveiller Process
- Surveiller Ping

## Zabbix Agent

```
wget https://repo.zabbix.com/zabbix/5.2/debian/pool/main/z/zabbix-  
release/zabbix-release_5.2-1+debian10_all.deb  
dpkg -i zabbix-release_5.2-1+debian10_all.deb  
apt update  
apt install zabbix-agent  
  
nano /etc/zabbix/zabbix_agentd.conf  
Server=10.0.0.213  
ServerActive=10.0.0.213  
Hostname=Server2  
service zabbix-agent restart
```

## Zabbix Agent & LXC Proxmox

Les containers LXC donnent de mauvaises valeurs avec l'agent Zabbix. Une solution existe, pour cela il faut demander à l'agent d'envoyer des données propres au container et créer un template Zabbix dédié pour ces hosts.

### Sur le host

1. Créer un fichier de configuration pour l'agent

```
nano /etc/zabbix/zabbix_agentd.d/zabbix_container.conf
```

Avec les données suivantes:

[zabbix\\_container.conf](#)

```
UserParameter=ct.memory.size[*],free -b | awk '$ 1 == "Mem:"  
{total=$ 2; used=($ 3+$ 5); pused=(( $ 3+$ 5)*100/$ 2); free=$ 4;  
pfree=($ 4*100/$ 2); shared=$ 5; buffers=$ 6; cached=$ 6;  
available=($ 7); pavailable=(( $ 7)*100/$ 2); if("$1" == "")  
{printf("%.0f", total )} else {printf("%.0f", $1 " )} }'  
UserParameter=ct.swap.size[*],free -b | awk '$ 1 == "Swap:"
```

```

{total=$ 2; used=$ 3; free=$ 4; pfree=($ 4*100/$ 2); pused=($
3*100/$ 2); if("$1" == "") {printf("%.0f", free )} else
{printf("%.0f", $1 "" )} }'
UserParameter=ct.cpu.load[*],uptime | awk -F'[ , ]+'
'{avg1=$(NF-2); avg5=$(NF-1); avg15=$(NF)}{print $2/'$(nproc)'}'
UserParameter=ct.uptime,cut -d"." -f1 /proc/uptime

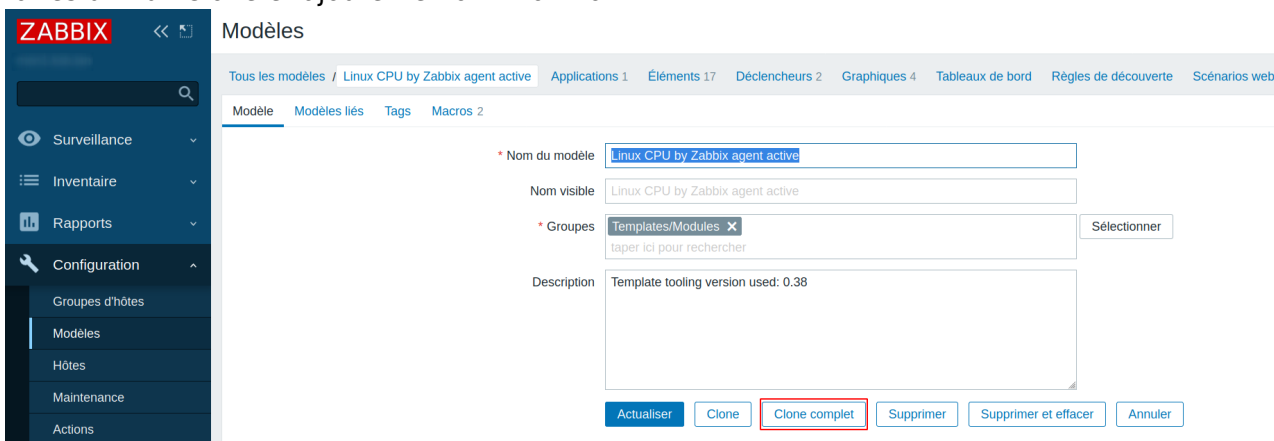
```

## 2. Relancer l'agent

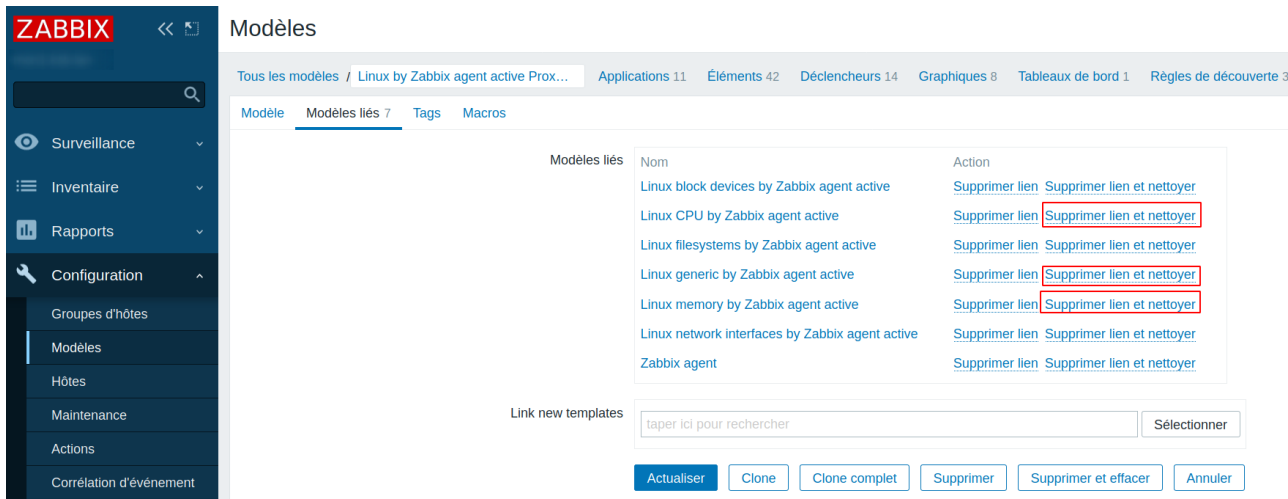
```
service zabbix-agent restart
```

## Sur le serveur Zabbix

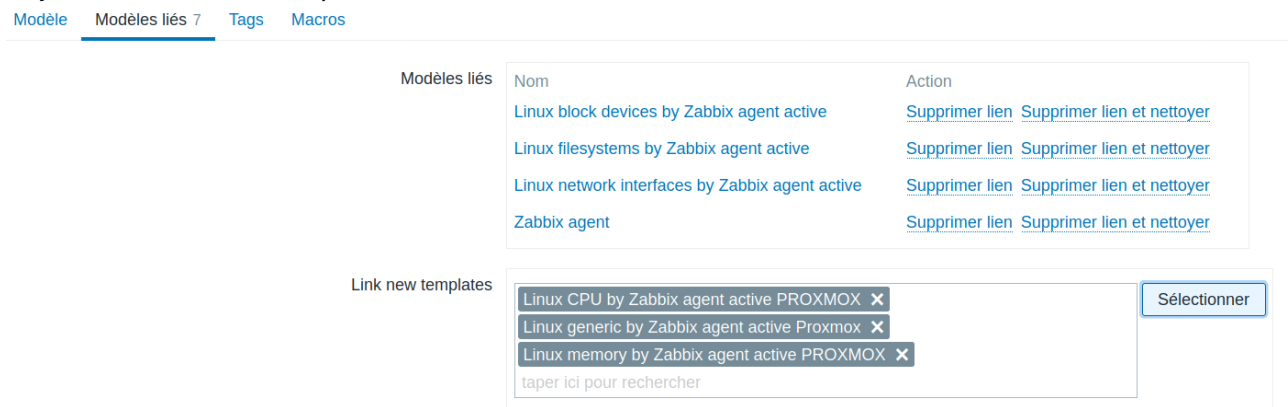
1. Dans le menu Configuration → Template, ouvrez le template 'Linux by Zabbix agent active' et faites un Full Clone et ajouter le nom Proxmox



2. Réouvrir “Linux CPU by Zabbix agent active PROXMOX” et se diriger dans les Items
3. Remplacer system.cpu.load[all,avgX] par ct.cpu.load[all,avg15]
4. A présent, ouvrir le template “Linux memory by Zabbix agent active” et faire un Full Clone et rajouter Proxmox
5. Réouvrir “Linux memory by Zabbix agent active PROXMOX” et se diriger dans les Items
6. Remplacer:
  1. vm.memory.size[available] par ct.memory.size[available]
  2. vm.memory.size[pavailable] par ct.memory.size[pavailable]
  3. system.swap.size[,free] par ct.swap.size[free]
  4. system.swap.size[,pfree] par ct.swap.size[pfree]
  5. vm.memory.size[total] par ct.memory.size[total]
  6. system.swap.size[,total] par ct.swap.size[total]
7. Ouvrir le template “Linux generic by Zabbix agent active” et faire un Full Clone et rajouter Proxmox dans le nom
8. Remplacer system.uptime par ct.uptime
9. Maintenant se rendre dans le template “Linux by Zabbix agent active”, faire un Full Clone en y rajoutant Proxmox dans le nom
10. Dans les linked templates, cliquer sur “Unlink & Clear” pour les templates associés CPU, Memory & Generic



### 11. Rajouter les 3 sous-templates créés



### 12. Affecter le nouveau template "Linux by Zabbix agent active Proxmox" aux machines concernées

## Single Sign On (SSO)

1. Installer les paquets kerberos nécessaire

```
apt-get -y install krb5-user libapache2-mod-auth-kerb
```

2. Editer le fichier krb5.conf

```
nano /etc/krb5.conf
```

Coller une configuration +/- similaire

krb5.conf

```
[libdefaults]
    default_realm = MAKEITSIMPLE.LAN

# The following krb5.conf variables are only for MIT Kerberos.
    kdc_timesync = 1
    ccache_type = 4
    forwardable = true
    proxiable = true
```

```
fcc-mit-ticketflags = true

[realms]
MAKEITSIMPLE.LAN = {
    kdc = srv-ad.makeitsimple.lan
    admin_server = srv-ad.makeitsimple.lan
    default_domain = srv-ad.makeitsimple.lan
}

.makeitsimple.lan = MAKEITSIMPLE.LAN
makeitsimple.lan = MAKEITSIMPLE.LAN
```

### 3. Faire un test de session

```
kinit Administrator
```

### 4. Se connecter sur le serveur Samba-AD et faire les 3 commandes suivantes en modifiant le nom du serveur

```
samba-tool user create --random-password http-glpi.makeitsimple.lan
samba-tool spn add HTTP/glpi.makeitsimple.lan http-
glpi.makeitsimple.lan
samba-tool domain exportkeytab /root/httpd.keytab --
principal=HTTP/glpi.makeitsimple.lan@MAKEITSIMPLE.LAN
```

### 5. Vérifier avec la commande kvno

```
kvno HTTP/glpi.makeitsimple.lan@MAKEITSIMPLE.LAN
```

### 6. De retour sur le serveur GLPI, déplacer le fichier dans le dossier apache et lui donner les bons droits

```
mv httpd.keytab /etc/apache2/
chown www-data:root /etc/apache2/httpd.keytab
chmod 640 /etc/apache2/httpd.keytab
```

### 7. Editer le fichier apache

```
nano /etc/apache2/sites-enabled/000-default.conf
```

#### 000-default.conf

```
DocumentRoot /var/www
ServerName glpi.makeitsimple.lan
<Location />
    AuthType Kerberos
    AuthName "Demande d'identification SSO"
    KrbAuthRealms MAKEITSIMPLE.LAN
    KrbServiceName HTTP/glpi.makeitsimple.lan
    Krb5Keytab /etc/apache2/httpd.keytab
```

```
KrbMethodNegotiate On
KrbMethodK5Passwd On
require valid-user
</Location>
```

## 8. Redémarrer apache2

```
Service apache2 restart
```

## 9. Ajouter le CA Certificate sur le serveur Zabbix

```
cp ca.crt /usr/local/share/ca-certificates/
update-ca-certificates<code> puis vérifier <code> bash>openssl s_client
-showcerts -connect dc1.makeitsimple.lan:636
```

, la réponse doit être "Verify return code: 0 (ok)"

## 1. Dans Zabbix -> Administration -> Authentification -> Ldap Parameters, faire la configuration pour se connecter à votre DC

## 2. Zabbix -> Administration -> Authentification -> Authentification -> Choisir LDAP

## 3. Dans Zabbix -> Administration -> Users créer des utilisateurs ayant le même login que sur le domaine. Les mots de passes ne sont pas nécessaires.

# Les noms de services Windows Dynamiques

Sous Windows, certains services sont générés de manière dynamique et apparaissent/disparaissent selon les besoins. Ceci crée de nouvelles alertes quand ils sont abandonnés par la machine. Pour ce faire, il faut faire une règle d'exclusion à rajouter dans Administration > General > Regular Expressions > Windows service names for discovery ceci

```
^\A(WpnUserService_|OneSyncSvc_|CDPUserService_)$
```

**ZABBIX** << [ ] Regular expressions v

Expressions Test

\* Name

\* Expressions

Expression type	Expression	Delimiter	Case sensitive	Action
Result is FALSE	^((MMCSS gupdate SysmonLog clr_optimization_v[0-9])		<input checked="" type="checkbox"/>	<a href="#">Remove</a>
Result is FALSE	^A(WpnUserService_OneSyncSvc_JCDPUserSv		<input type="checkbox"/>	<a href="#">Remove</a>

[Add](#)

## Debugger snmpv3

```
apt install snmp
snmpwalk -v3 -a MD5 -A dS2N9xjf0j5TlbH -u ciscosnmp -x DES -X
dS2N9xjf0j5TlbH -l authPriv 10.10.10.41
```

From:  
<https://wiki.makeitsimple.be/> - makeITsimple wiki

Permanent link:  
<https://wiki.makeitsimple.be/doku.php?id=linux:zabbix&rev=1615979641>

Last update: **2021/06/20 09:42**

