

ZABBIX

Superviser les services

Thomas Calatayud

Dans le cadre de la supervision du SAEiR, il m'a été demandé de superviser certains services applicatifs s'exécutant sur les serveurs. Pour l'exemple nous allons superviser une application antivirus qui s'exécute en tant que service système linux. Cet antivirus est identifier par le système par : clamd.

Solutions

UserParameters

Le *UserParameter* est un paramètre propre à zabbix. Il permet d'exécuter n'importe quelle commande depuis un terminal par le biais de l'agent zabbix. L'agent zabbix peut récupérer le résultat de la commande exécuté et l'envoyer au proxy ou serveur pour y accéder sur l'interface de zabbix et ainsi pouvoir créer des items et triggers afin d'être notifié des problèmes lié à l'application en question.

Utilisation

La première étape est d'identifier la commande qui va nous donner les informations sur le statut du service. Ici, nous utilisons la commande service fournit par linux.

```
tcalatayud@tcalatayud-CD49:~$ service clamd status  
clamd (pid 1309) is running... ①
```

① Nous pouvons voir ici que le service s'exécute correctement avec le pid 1309.

Maintenant que nous avons identifié la commande a utiliser il va falloir l'intégrer au fichier de configuration de l'agent zabbix /etc/zabbix/zabbix_agent.conf. Nous rajoutons à la fin du fichier :

```
...  
UserParameter=service.check[*],service $1 status ①
```

① Nous indiquons ici que nous allons utiliser un UserParameter en précisant la clé d'identification : *service.check[*]* permettant d'identifier la commande *service \$1 status* qui sera exécuté.

Après avoir redémarrer l'agent zabbix nous pouvons créer dans l'interface un nouvel item qui sera associé à ce UserParameter.

Dans l'onglet Configuration > Hosts sélectionner *Items* de l'hôte voulu.

Choisir l'option *Create item*

Superviser les services -

The screenshot shows the Zabbix Server Configuration of Items page. The form is for creating a new item. The fields are as follows:

- Name: Service clamd status
- Type: Zabbix agent (active)
- Key: service.check[clamd]
- Type of information: Text
- Update interval: 30s
- History storage period: 90d
- New application: (empty)
- Applications: (list of applications including CPU, Filesystems, General, Memory, Network interfaces, OS, Performance, Processes, Security, Service check, Zabbix agent active)
- Populates host inventory field: -None-
- Description: (empty)
- Enabled: ☒
- Buttons: Add, Cancel

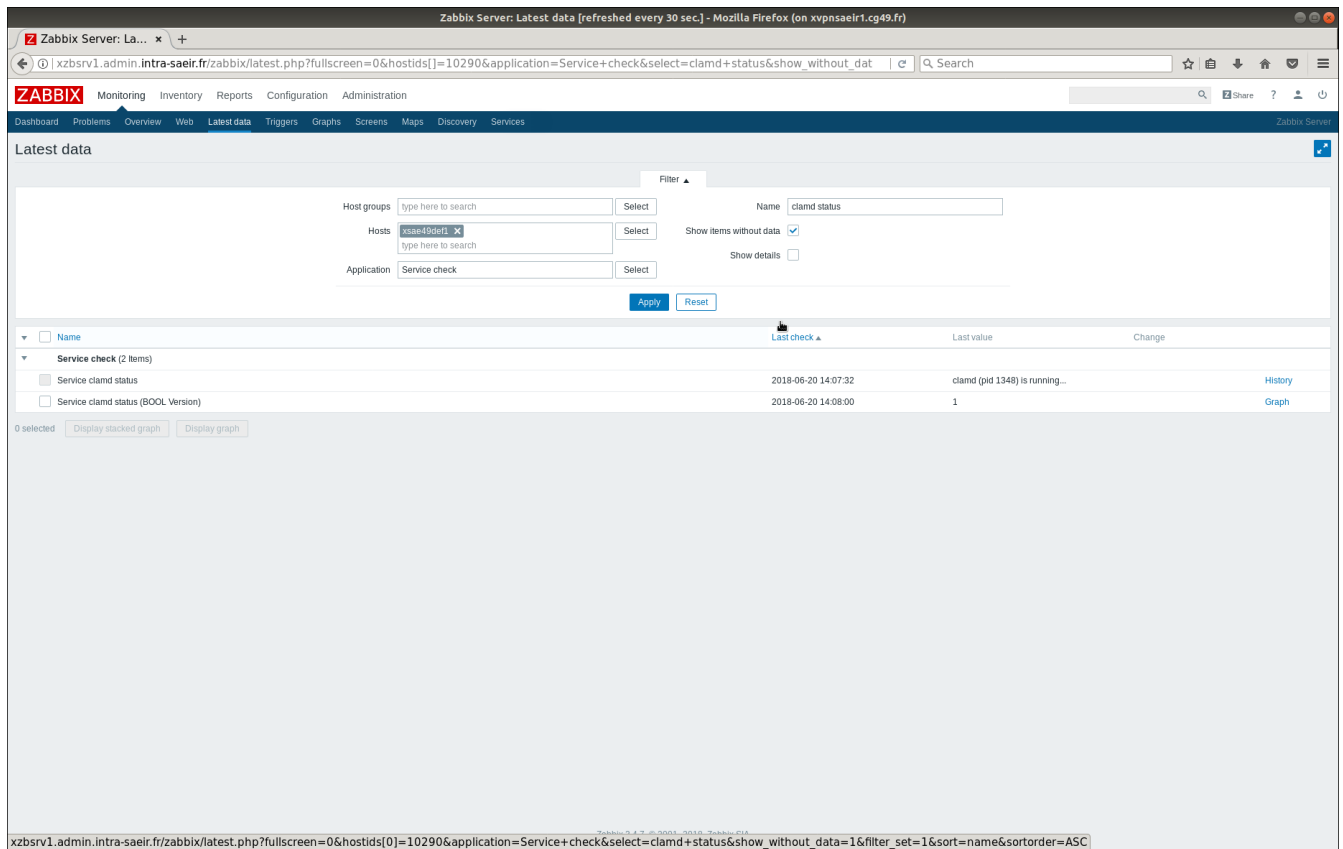
1. Préciser le nom de l'item.
2. Choisir si l'agent zabbix est actif ou passif.
3. Indiquer la clé du UserParameter que nous avons spécifié dans le fichier de configuration de l'agent zabbix en précisant le paramètre entre crochet. Ici, le paramètre correspond au nom du service et remplacera dans la commande `service $1 status` le \$1.
4. Bien définir le type d'information, ici le résultat de la commande nous donne l'information sous forme d'une chaîne de caractère donc nous choisissons l'option `Text`.
5. Nous pouvons choisir les options de rétention et d'historisation.
6. Ne pas oublier de préciser l'application qui sera lié à cet item pour permettre d'identifier plus simplement l'item ou en créer une nouvelle.
7. Appuyer sur le bouton `Add` pour finaliser la création de l'item.

Nous avons créé un item qui nous indique si le service est bien lancé ou non. Cependant, cet item nous renvoie une chaîne de caractère difficile à évaluer par un trigger nous allons donc transformer la réponse de cet item en entier qui sera beaucoup plus facile à évaluer. Si le service tourne il nous retournera 1 sinon il retournera 0. Pour modifier cet item revoyons le UserParameter. Nous allons rajouter toujours à la fin de notre fichier de configuration de l'agent zabbix.

```
...
UserParameter=service.check[*],service $1 status ①
UserParameter=service.check-int[*],(service $1 status && echo "1" || echo "0") | sed
-n 2p
```

Toujours de la même façon, nous allons recréer un nouvel item en définissant correctement le type d'information *Numeric (unsigned)*.

on obtient ainsi les deux items suivants :



The screenshot shows the Zabbix Server interface in a Mozilla Firefox browser. The page title is 'Zabbix Server: Latest data [refreshed every 30 sec.]'. The URL is 'xzbsrv1.admin.intra-saeir.fr/zabbix/latest.php?fullscreen=0&hostids[]=10290&application=Service+check&select=clamd+status&show_without_data=1&filter_set=1&sort=name&sortorder=ASC'. The page displays a table of latest data for the 'Service clamd status' item. The table has columns for 'Name', 'Last check', 'Last value', and 'Change'. There are two items listed: 'Service clamd status' and 'Service clamd status (BOOL Version)'. The first item has a last check of '2018-06-20 14:07:32' and a last value of 'clamd (pid 1348) is running...'. The second item has a last check of '2018-06-20 14:08:00' and a last value of '1'. The page also includes a filter section at the top with fields for 'Host groups', 'Hosts', 'Application', and 'Name', and buttons for 'Apply' and 'Reset'.

Name	Last check	Last value	Change
Service clamd status	2018-06-20 14:07:32	clamd (pid 1348) is running...	History
Service clamd status (BOOL Version)	2018-06-20 14:08:00	1	Graph

Maintenant que l'item est créé nous allons créer un trigger qui lui sera associé et qui nous permettra de lever des alertes au cas où le service serait coupé. Pour cela nous devons retourner sur la page de sélection de l'hôte (Configuration > Hosts). Ici, nous sélectionnons *Triggers* sur l'hôte en question.

Zabbix Server: Configuration of hosts - Mozilla Firefox (on xvpnsaeir1.cg49.fr)

URL: xzbsrv1.admin.intra-saeir.fr/zabbix/hosts.php?hostid=10289&groupid=23

ZABBIX Monitoring Inventory Reports Configuration Administration

Host groups Templates Hosts Maintenance Actions Event correlation Discovery Services

Hosts Group: Serveur prod-int Create host Import

Filter Name DNS IP Port Apply Reset

Name	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Templates	Status	Availability	Agent encryption	Info
xzbpndre1:xsae49int1	Applications 11	Items 58	Triggers 24	Graphs 11	Discovery 2	Web	172.20.49.2:10050	Template OS Linux Active (Template App Zabbix Agent Active), Template Services Check	Enabled	ZBX SNMP JMX IPMI	NONE	
xzbpndre1:xsae53int1	Applications 11	Items 58	Triggers 23	Graphs 11	Discovery 2	Web	172.20.53.2:10050	Template OS Linux Active (Template App Zabbix Agent Active), Template Services Check	Enabled	ZBX SNMP JMX IPMI	NONE	
xzbpndre1:xsae72int1	Applications 11	Items 58	Triggers 23	Graphs 11	Discovery 2	Web	172.20.72.2:10050	Template OS Linux Active (Template App Zabbix Agent Active), Template Services Check	Enabled	ZBX SNMP JMX IPMI	NONE	
xzbpndre1:xsae85int1	Applications 11	Items 58	Triggers 23	Graphs 11	Discovery 2	Web	172.20.85.2:10050	Template OS Linux Active (Template App Zabbix Agent Active), Template Services Check	Enabled	ZBX SNMP JMX IPMI	NONE	
xzbpndre1:xsae99int1	Applications 11	Items 58	Triggers 23	Graphs 11	Discovery 2	Web	172.20.99.2:10050	Template OS Linux Active (Template App Zabbix Agent Active), Template Services Check	Enabled	ZBX SNMP JMX IPMI	NONE	

Displaying 5 of 5 found

0 selected Enable Disable Export Mass update Delete

xzbsrv1.admin.intra-saeir.fr/zabbix/triggers.php?groupid=23&hostid=10289

Zabbix 3.4.7 © 2001-2018, Zabbix SIA

Choisir l'option *Create trigger*

Zabbix Server: Configuration of triggers - Mozilla Firefox (on xvpnsaeir1.cg49.fr)

URL: xzbsrv1.admin.intra-saeir.fr/zabbix/triggers.php?groupid=23&hostid=10289

ZABBIX Monitoring Inventory Reports Configuration Administration

Host groups Templates Hosts Maintenance Actions Event correlation Discovery Services

Triggers Group: Serveur prod-int Host: xsae49int1 Create trigger

All hosts / xzbpndre1:xsae49int1 Enabled ZBX SNMP JMX IPMI Applications 11 Items 58 Triggers 24 Graphs 11 Discovery rules 2 Web scenarios

Filter Severity State Status Apply Reset

Severity: all Not classified Information Warning Average High Disaster

State: all Normal Unknown

Status: all Enabled Disabled

Severity	Name	Expression	Status	Info
Warning	Template OS Linux Active: /etc/passwd has been changed on (HOSTNAME)	{xsae49int1.vfs.file.cksum[/etc/passwd].diff(0)}>0	Enabled	
Warning	Template Services Check: Clamd is down	{xsae49int1.service.check-in[clamd].last(0)}=0	Enabled	
Information	Template OS Linux Active: Configured max number of opened files is too low on (HOSTNAME)	{xsae49int1.kernel.maxfiles.last(0)}<1024	Enabled	
Information	Template OS Linux Active: Configured max number of processes is too low on (HOSTNAME)	{xsae49int1.kernel.maxproc.last(0)}<256	Enabled	
Warning	Template OS Linux Active: Disk I/O is overloaded on (HOSTNAME)	{xsae49int1.system.cpu.util[10min].avg(5m)}>20	Enabled	
Average	Mounted filesystem discovery: Free disk space is less than 20% on volume /	{xsae49int1.vfs.fs.size[/].pfree}.last(0)<20	Enabled	
Average	Mounted filesystem discovery: Free disk space is less than 20% on volume /boot	{xsae49int1.vfs.fs.size[/boot.pfree].last(0)}<20	Enabled	
Average	Mounted filesystem discovery: Free disk space is less than 20% on volume /data/archives-01	{xsae49int1.vfs.fs.size[data/archives-01.pfree].last(0)}<20	Unknown	!
Average	Mounted filesystem discovery: Free disk space is less than 20% on volume /data/archives-02	{xsae49int1.vfs.fs.size[data/archives-02.pfree].last(0)}<20	Unknown	!
Average	Mounted filesystem discovery: Free inodes is less than 20% on volume /	{xsae49int1.vfs.fs.inode[/].pfree}.last(0)<20	Enabled	
Average	Mounted filesystem discovery: Free inodes is less than 20% on volume /boot	{xsae49int1.vfs.fs.inode[/boot.pfree].last(0)}<20	Enabled	
Average	Mounted filesystem discovery: Free inodes is less than 20% on volume /data/archives-01	{xsae49int1.vfs.fs.inode[data/archives-01.pfree].last(0)}<20	Unknown	!
Average	Mounted filesystem discovery: Free inodes is less than 20% on volume /data/archives-02	{xsae49int1.vfs.fs.inode[data/archives-02.pfree].last(0)}<20	Unknown	!
Information	Template OS Linux Active: Host information was changed on (HOSTNAME)	{xsae49int1.system.uptime.diff(0)}>0	Enabled	
Information	Template App Zabbix Agent Active: Host name of zabbix_agentd was changed on (HOSTNAME)	{xsae49int1.agent.hostname.diff(0)}>0	Enabled	
Information	Template OS Linux Active: Hostname was changed on (HOSTNAME)	{xsae49int1.system.hostname.diff(0)}>0	Enabled	
Average	Template OS Linux Active: Lack of available memory on server (HOSTNAME)	{xsae49int1.vm.memory.size[available].last(0)}<20M	Enabled	
Warning	Template OS Linux Active: Lack of free swap space on (HOSTNAME)	{xsae49int1.system.swap.size.pfree}.last(0)<50	Enabled	
Warning	Template OS Linux Active: Processor load is too high on (HOSTNAME)	{xsae49int1.system.cpu.load[percpu.avg1].avg(5m)}>5	Enabled	
High	TEST Disk I/O is overloaded on (HOSTNAME) TEST	{xsae49int1.system.cpu.util[10min].min(10m)}>20	Enabled	
Warning	Template OS Linux Active: Too many processes on (HOSTNAME)	{xsae49int1.proc.num.avg(5m)}>300	Enabled	
Warning	Template OS Linux Active: Too many processes running on (HOSTNAME)	{xsae49int1.proc.num[run].avg(5m)}>30	Enabled	
Information	Template App Zabbix Agent Active: Version of zabbix_agentd was changed on (HOSTNAME)	{xsae49int1.agent.version.diff(0)}>0	Enabled	

Zabbix Server: Configuration of triggers - Mozilla Firefox (on xvpnsaeir1.cg49.fr)

http://xzbsrv1.admin.intra-saeir.fr/zabbix/triggers.php?groupid=236&hostid=102896&form=Create+trigger

ZABBIX Monitoring Inventory Reports Configuration Administration

Host groups Templates Hosts Maintenance Actions Event correlation Discovery Services

Triggers

All hosts / xzbsrv1:xsae49int1 Enabled [ZBX] [SNMP] [JMX] [IPMI] Applications 11 Items 58 Triggers 24 Graphs 11 Discovery rules 2 Web scenarios

Trigger Dependencies

Name Clamd is down

Severity Not classified Information **Warning** Average High Disaster

Expression (xsae49int1.service.check-inf[clamd].last())=0 [Add](#)

[Expression constructor](#)

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Tags tag value [Remove](#)

[Add](#)

Allow manual close ☐

URL

Description

Enabled ☒

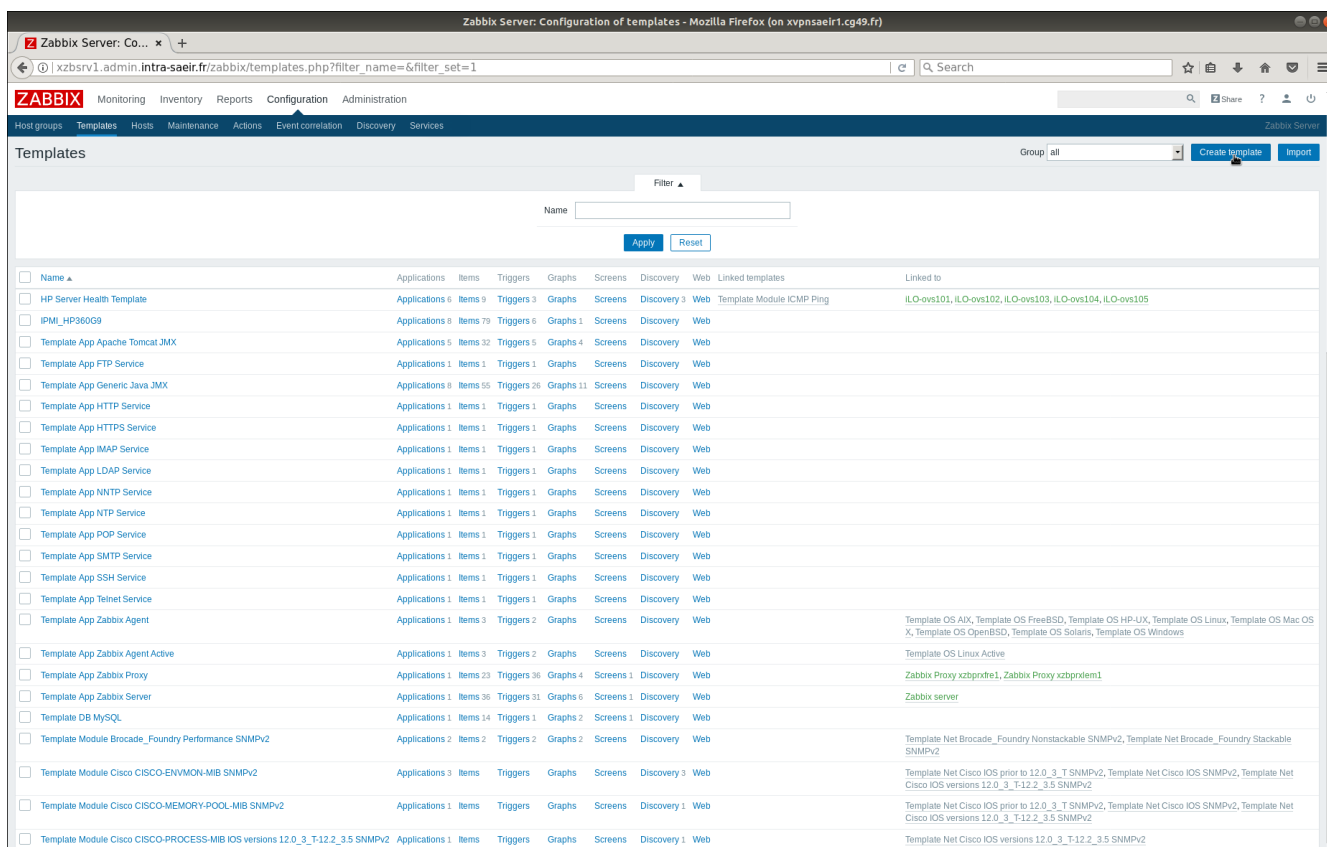
[Add](#) [Cancel](#)

Zabbix 3.4.7 © 2001–2018, Zabbix SIA

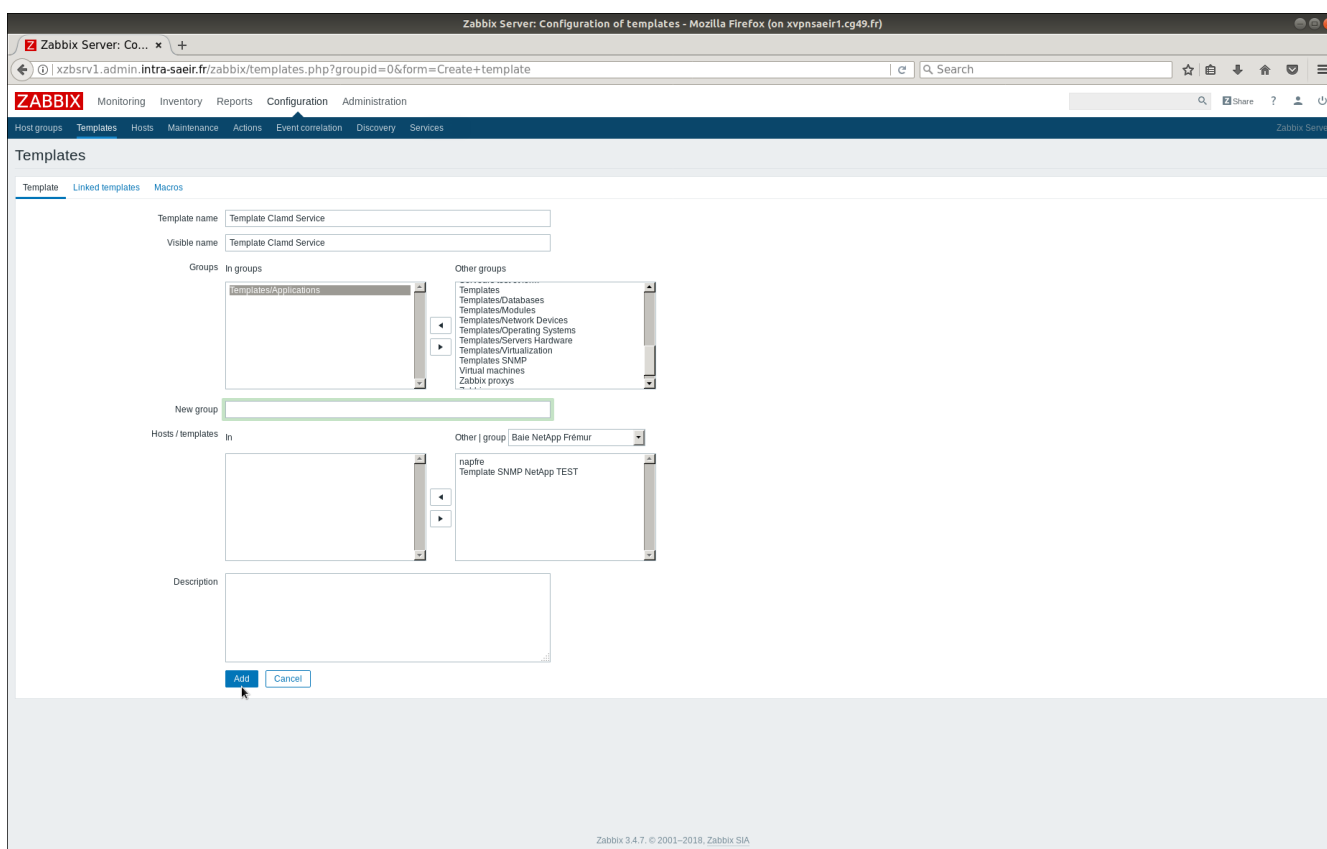
1. Préciser le nom du trigger.
2. Indiquer sa sévérité.
3. Construire l'expression qui déclenchera le trigger. Pour cela, inscrire directement l'expression dans le champs ou appuyer sur le bouton *Add*.

- a. Sélectionner l'item lié au trigger.
 - b. Indiquer la fonction. Ici on veut déclencher le trigger si la dernière valeur relevé par zabbix est 0. Pour cela on prend la fonction Las(most recent) T value is = N, où T est notre dernière valeur en question et N notre valeur 0 a indiquer dans la case N.
 - c. Appuyer sur le bouton *Insert* qui va rajouter l'expression automatiquement dans notre trigger.
4. Finaliser la création du trigger en appuyant sur le bouton *Add*

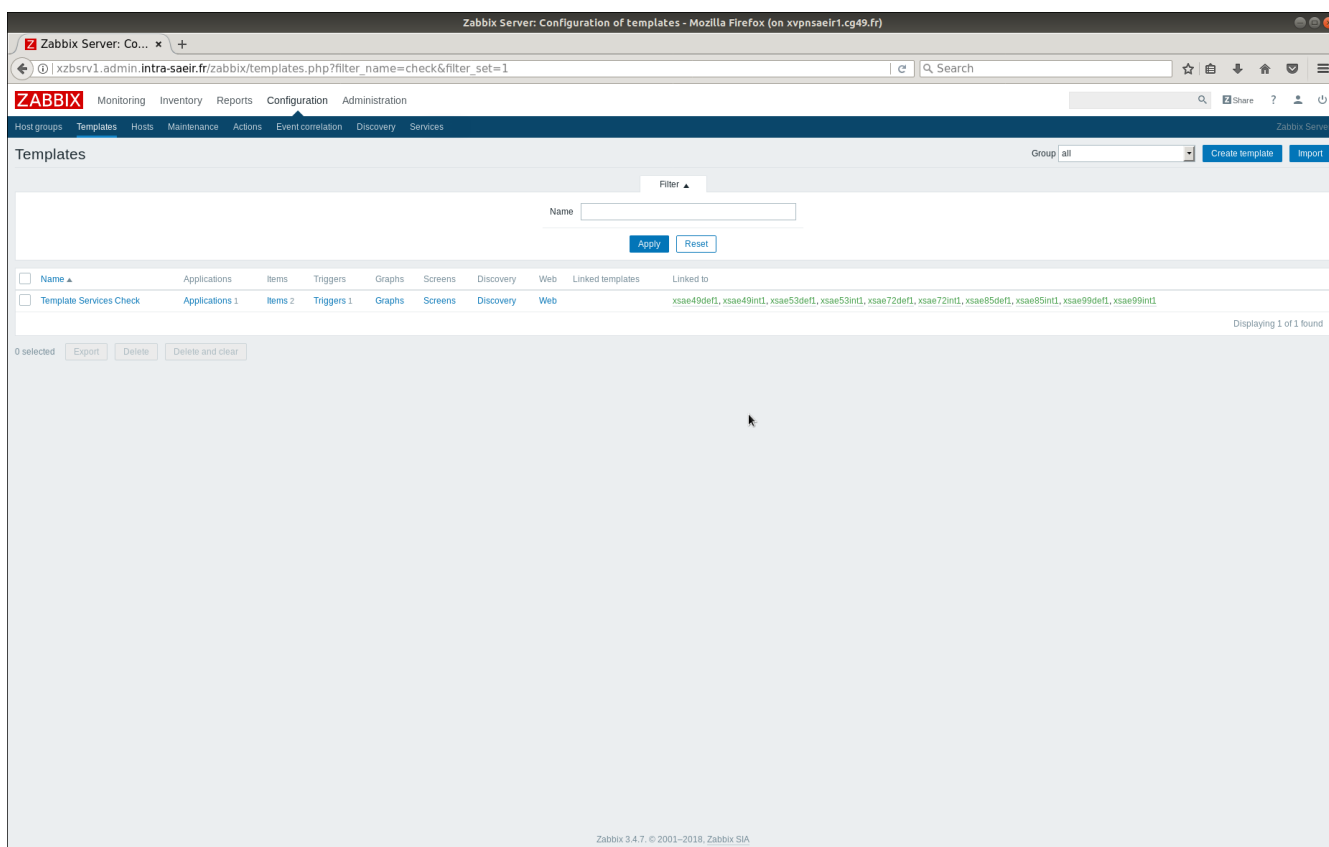
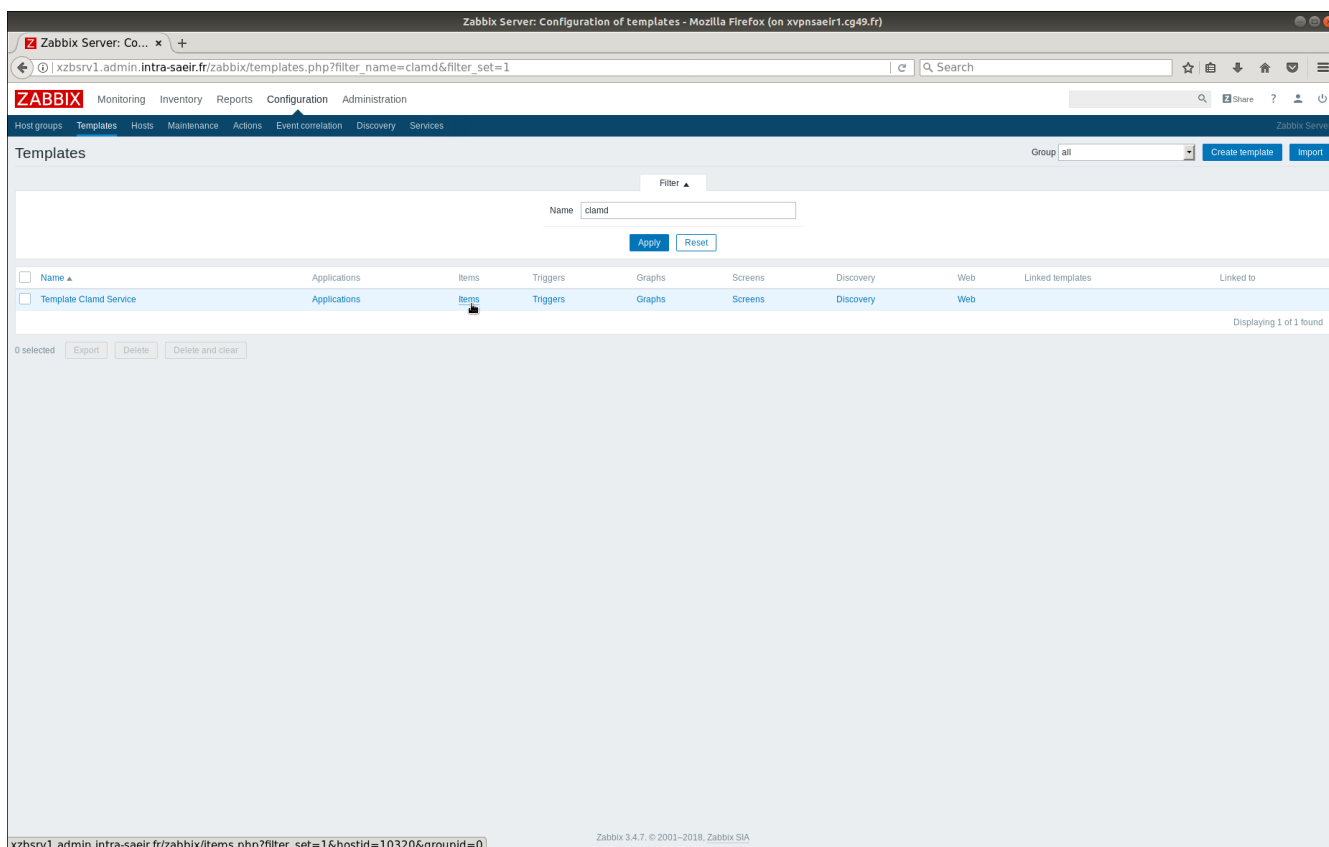
Nous avons maintenant notre service d'antivirus totalement supervisé sur l'hôte que nous souhaitons. Cependant, il pourrait être intéressant si nous voulons déployer ce couple d'item et de trigger sur plusieurs hôtes de créer un template regroupant les deux. Avec ce template la supervision du service serait déployable sur n'importe quel hôte. Pour cela il suffit de créer un nouveau template dans Configuration > Templates en cliquant sur le bouton *Create template*.



On spécifie le nom du template et on crée le template en cliquant sur le bouton *Add* et en ne pas oubliant de lui attribuer le groupe qui lui correspond ou en créer un nouveau.



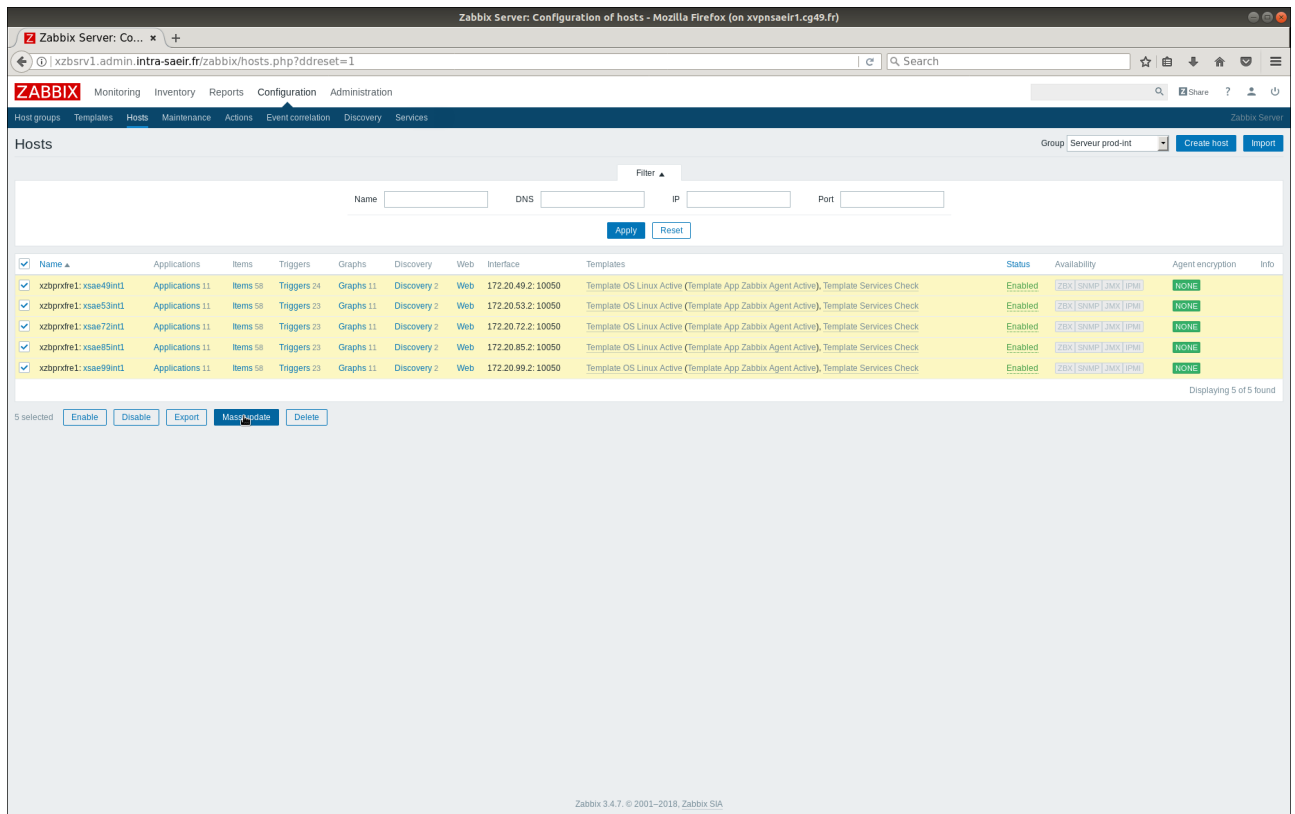
Une fois le template créé il reste plus qu'à lui rajouter les items et triggers souhaités de la même méthode expliqué plus haut pour les hôtes.



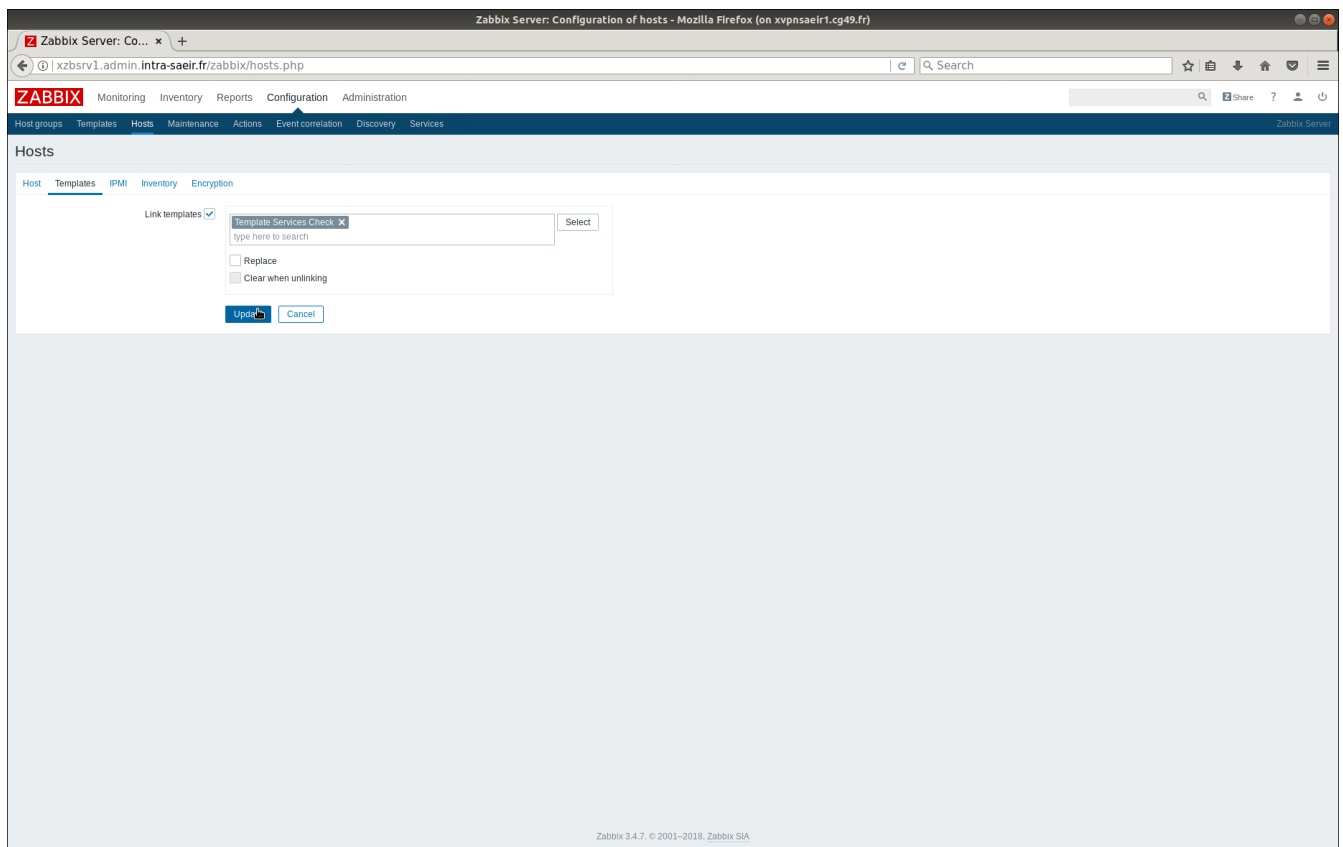
Il ne reste plus qu'à assigné le templates à tous les hôtes nécessaires. Ici nous allons comment assigner rapidement le template à des hôtes déjà existant.

Pour cela, dans l'onglet Configuration > Hosts

1. Sélectionner tous les hôtes et cliquer sur le bouton *Mass update*



2. Dans l'onglet *Templates*, indiquer le template à ajouter.
3. Cliquer sur le bouton *Update* pour mettre à jour les hôtes.



Zabbix supervise maintenant les services antivirus de tous nos serveurs.

Host	Name	Last check	Last value	Change
xsae49def1	Service check (2 items)			
	Service clamd status (BOOL Version)	2018-06-20 14:55:01	1	Graph
	Service clamd status	2018-06-20 14:55:04	clamd (pid 1348) is running...	History
xsae53def1	Service check (2 items)			
	Service clamd status (BOOL Version)	2018-06-20 14:54:54	1	Graph
	Service clamd status	2018-06-20 14:55:09	clamd (pid 1347) is running...	History
xsae99def1	Service check (2 items)			
	Service clamd status	2018-06-20 14:55:02	clamd (pid 1361) is running...	History
	Service clamd status (BOOL Version)	2018-06-20 14:55:02	1	Graph
xsae85def1	Service check (2 items)			
	Service clamd status	2018-06-20 14:54:53	clamd (pid 1374) is running...	History
	Service clamd status (BOOL Version)	2018-06-20 14:55:02	1	Graph
xsae72def1	Service check (2 items)			
	Service clamd status	2018-06-20 14:54:55	clamd (pid 1348) is running...	History
	Service clamd status (BOOL Version)	2018-06-20 14:55:09	1	Graph
xsae49int1	Service check (2 items)			
	Service clamd status	2018-06-20 14:54:56	clamd (pid 1330) is running...	History
	Service clamd status (BOOL Version)	2018-06-20 14:55:06	1	Graph
xsae53int1	Service check (2 items)			
	Service clamd status (BOOL Version)	2018-06-20 14:54:51	1	Graph
	Service clamd status	2018-06-20 14:55:01	clamd (pid 18626) is running...	History
xsae72int1	Service check (2 items)			
	Service clamd status	2018-06-20 14:54:48	clamd (pid 1383) is running...	History
	Service clamd status (BOOL Version)	2018-06-20 14:55:10	1	Graph

Conclusion

Pendant ce tutoriel nous avons appris à utiliser les UserParameter de zabbix dans le cadre de la supervision du service clamd. Mais, il est bien entendu possible et très simple d'adapté cette méthode pour superviser n'importe quel service en modifiant simplement lors de la création de l'item le paramètre entre [] de la clé du UserParameter. Il est aussi possible très simplement en modifiant dans le fichier de configuration zabbix la commande du UserParameter, de superviser n'importe quelle autre application à partir du moment où elle peut afficher son statut en ligne de commande.

Nous avons aussi appris à créer des items, des triggers et les inclure à un template pour le déployer sur plusieurs hôtes très rapidement.