Table des matières

1.	Relancer le processus Apache sous Linux	. 1
	and the second production of	
2.		. 5

1. Relancer le processus Apache sous Linux

Pour pouvoir relancer le processus Apache automatiquement s'il tombe nous devons créer une commande qui va relancer le service manuellement si notre serveur centreon remarque un état critique sur notre service de check apache

Pour se faire nous devons nous connecter au serveur centreon avec l'utilisateur centreon-engine :

su - centreon-engine

Puis générer une paire de clé publique/privée ssh :

ssh-keygen

Et copier la clé publique sur notre machine debian qui tient le serveur Apache :

ssh-copy-id -i \$USER\$@<IP Serveur Debian>

/!\Attention à bien remplacer le \$USER\$ par le nom d'utilisateur de la machine debian et l'IP par son IP

Maintenant nous devons aller sur notre machine debian et éditer le fichier /etc/sudoers.d/\$USER\$:

/!\ Attention à bien remplacer le \$USER\$ par le nom d'utilisateur de la machine debian

Debian Engine

\$USER\$ ALL = NOPASSWD: /usr/sbin/service apache2 restart

Editer ce fichier va permettre de ne pas donner le mot de passe lors du redémarrage du service apache par l'utilisateur

Nous pouvons maintenant tester une commande qui vérifie l'état du service Apache :

#APACHE ETEINT

[centreon-engine@localhost plugins]\$ /usr/local/nagios/libexec/check_by_ssh -H 192.168.7.1 <mark>-i</mark> /var/lib/centreon-engine/.ssh/id_rsa -l debian -C '/usr/local/nagios/libexec/check_procs -a '/usr/sbin/apache2' -c 5:1024' -E

PROCS CRITICAL: 0 processes with args '/usr/sbin/apache2' | procs=0;;5:1024;0;

#APACHE ALLUME

[centreon-engine@localhost plugins]\$ /usr/local/nagios/libexec/check_by_ssh -H 192.168.7.1 -i /var/lib/centreon-engine/.ssh/id_rsa -l debian -C '/usr/local/nagios/libexec/check_procs -a '/usr/sbin/apache2' -c 5:1024' -E

ROCS OK: 6 processes with args '/usr/sbin/apache2' | procs=6;;5:1024;0;

- -c : Défini le seuil critique (ici moins de 5 processus apache on passe en état critique)
- -i : Défini le chemin vers la clé privée du serveur centreon
- -l : Défini le nom d'utilisateur sur la machine à laquelle on se connecte en SSH
- -C: Défini la commande que l'on passe sur la machine

Une fois que l'on sait si notre service Apache fonctionne on peut lancer une commande qui redémarre le service s'il tombe :

On voit bien qu'il ne nous a pas demandé de mot de passe pour le redémarrage du service

Passage des commandes dans centreon :

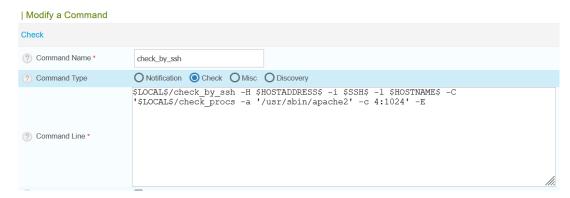
Maintenant que nous avons des commandes fonctionnelles nous pouvons les mettre en place dans centreon

Tout d'abord il faut créer les commandes :

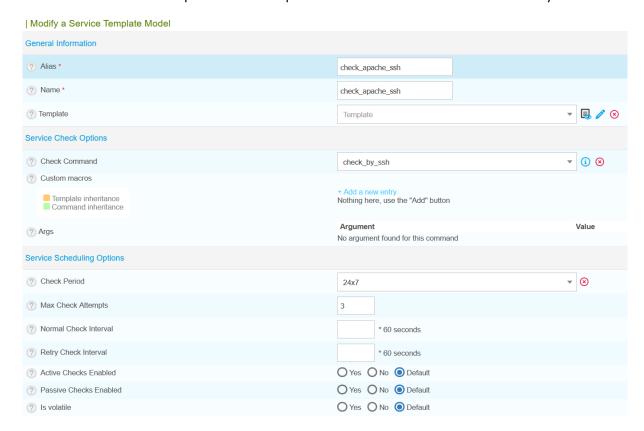
Commande en Event Handler:



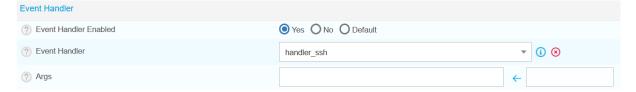
Commande de check by SSH:



Ensuite il faut créer un template de service qui va être lié à notre commande de check by ssh :

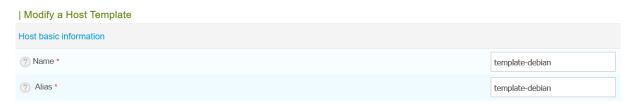


Dans l'onglet "Data Processing" nous devons activer l'event handler et y associer notre commande créée précédemment :

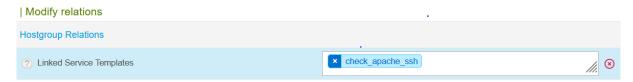


/!\ Attention : Si à la fin de la manipulation le service ne remonte pas automatiquement il se peut qu'il faille faire la manipulation juste au-dessus dans l'onglet Data Processing du service by host

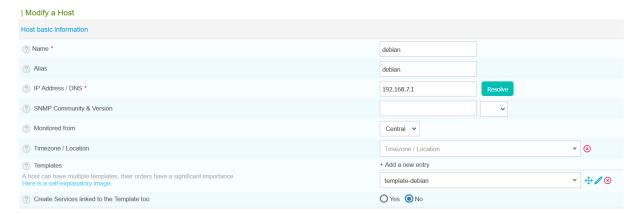
Nous devons maintenant créer un template d'hôte et le lier à notre template de service :



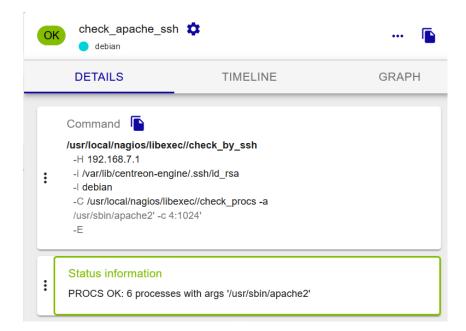
La liaison au template de service se fait dans l'onglet relations :



Il ne nous reste plus qu'à lier notre template d'hôte à notre hôte debian :



Nous devons exporter la configuration puis nous rendre dans l'onglet Monitoring pour procéder aux tests :



Notre service apache est fonctionnel nous pouvons maintenant le "tuer" :

root@debian10:/usr/lib/centreon/plugins# pkill apache2

On voit bien que le service n'est plus en "active running"

```
root@debian10:/usr/lib/centreon/plugins# systemctl status apache2

• apache2.service - The Apache HTTP Server

Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)

Active: deactivating (stop) since Thu 2022-01-13 09:52:09 CST; 2s ago

Docs: https://httpd.apache.org/docs/2.4/

Process: 6133 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)

Main PID: 6137 (code=exited, status=0/SUCCESS); Control PID: 6460 (apachectl)

Tasks: 2 (limit: 4915)
```





2.