

Supervision

I-Ajouter une machine hôte à la supervision

Sur votre serveur Linux à surveiller...

La procédure est un peu plus longue. Il faut d'abord installer le daemon NRPE et les plugins Nagios (qui vont être lancés localement par le daemon NRPE):

Sous Ubuntu/Debian:

```
# sudo apt-get install nagios-nrpe-server
# sudo apt-get install nagios-plugins
# sudo apt-get install xinetd
#apt-get install check-mk-agent
```

Puis éditer le fichier /etc/nagios/nrpe.cfg pour modifier la ligne suivante:

```
...
allowed_hosts = Mettre ici l'adresse IP de votre serveur Nagios
...
```


On ajoute une règle pour autoriser le Firewall IPtable à laisser passer les requêtes NRPE (à adapter selon vos règles):

```
root@racktables-server:/etc/xinetd.d# iptables -A INPUT -p tcp -m tcp --dport 5666 -j ACCEPT

root@racktables-server:/etc/xinetd.d# iptables -L
Chain INPUT (policy ACCEPT)
target prot opt source destination
ACCEPT tcp -- anywhere anywhere tcp dpt:nrpe

Chain FORWARD (policy ACCEPT)
target prot opt source destination

Chain OUTPUT (policy ACCEPT)
target prot opt source destination
```

// Vérifier si la machine écoute bien dans le port 5666 avec la commande ci-dessous :

```
root@racktables-server:/etc/xinetd.d# netstat -tulpn
Connexions Internet actives (seulement serveurs)
Proto Recv-Q Send-Q Adresse locale Adresse distante Etat PID/Program name
tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN 6551/sshd
tcp 0 0 127.0.0.1:25 0.0.0.0:* LISTEN 15119/exim4
tcp 0 0 0.0.0.0:6556 0.0.0.0:* LISTEN 13149/xinetd
tcp 0 0 0.0.0.0:5666 0.0.0.0:* LISTEN 13341/nrpe
tcp 0 0 0.0.0.0:111 0.0.0.0:* LISTEN 1808/rpcbind
tcp6 0 0 :::22 :::* LISTEN 6551/sshd
tcp6 0 0 :::1:25 :::* LISTEN 15119/exim4
tcp6 0 0 :::443 :::* LISTEN 4648/apache2
tcp6 0 0 :::5666 :::* LISTEN 13341/nrpe
```

Etat de la machine racktables-server dans la supervision avant les manipes:

State	Service	Icons	Status detail	Age	Checked	Perf-O-Meter
CRIT	Check_MK		CRIT - Cannot get data from TCP port 172.31.131.37:6556: [Errno 111] Connection refused, execution time 0.0 sec	17 sec	17 sec	0.0 s
WARN	Check_MK Inventory		WARN - 2 unchecked services (ps:2)	2017-05-10 14:26:47	19 min	
OK	[SYS] CPU utilisee		OK - user: 0.3%, system: 0.1%, wait: 0.0%, total: 0.5%	14 min	5 min	0%
OK	[SYS] FS /		OK - 11.1% used (6.29 of 56.49 GB), (levels at 80.00/90.00%), trend: +8.30 kB / 24 hours, inodes available 3687k/97.63%	14 min	5 min	11.14 %
UNKN	[SYS] FS /boot		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /home		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /opt		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /utils		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /tmp		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /usr		UNKNOWN - filesystem not found	14 min	5 min	
UNKN	[SYS] FS /var		UNKNOWN - filesystem not found	14 min	5 min	
OK	[SYS] IO Disque SUMMARY		OK - 575.74 B/sec read, 3.58 kB/sec write, IOs: 0.42/sec	9 min	5 min	0.00 M/s 0.00 M/s
OK	[SYS] Kernel Context Switches		OK - 84/s	9 min	5 min	83.6/s
OK	[SYS] Kernel Major Page Faults		OK - 0/s	9 min	5 min	0.0/s
OK	[SYS] Kernel Process Creations		OK - 1/s	9 min	5 min	0.6/s
OK	[SYS] Load		OK - 15min load 0.05	14 min	5 min	0.0
OK	[SYS] Memoire utilisee		OK - 0.41 GB used (0.40 RAM + 0.00 SWAP + 0.01 Pagetables, this is 20.7% of 1.96 RAM (2.47 total SWAP)), 0.0 mapped, 0.8 committed, 0.0 shared	14 min	5 min	20%
OK	[SYS] Nombre de Threads		OK - 107 threads	14 min	5 min	107
OK	[SYS] Process RSYSLOG		OK - 1 processes 252.6 MB virtual, 3.7 MB resident, 0.0% CPU	14 min	5 min	0.0%
OK	[SYS] Process SYSLOG		OK - 1 processes 252.6 MB virtual, 3.7 MB resident, 0.0% CPU	14 min	5 min	0.0%

Maintenant, lancez le service nagios-nrpe avec la commande suivante:

```
systemctl restart nagios-nrpe-server.service
systemctl restart xinetd.service
```

Dans certains distributions, il faut activer le script de lancement de xinetd. Sinon le service xinetd ne sera pas lancé automatiquement lors du prochain reboot.

On RedHat, SUSE and FEDORA this is done by:

```
root@racktables-server:/# chkconfig xinetd on
```

```

root@racktables-server:/# netstat -ltn
Connexions Internet actives (seulement serveurs)
Proto Recv-Q Send-Q Adresse locale      Adresse distante     Etat
tcp      0      0 0.0.0.0:22          0.0.0.0:*            LISTEN
tcp      0      0 127.0.0.1:25        0.0.0.0:*            LISTEN
tcp      0      0 0.0.0.0:6556        0.0.0.0:*            LISTEN
tcp      0      0 0.0.0.0:5666        0.0.0.0:*            LISTEN
tcp      0      0 0.0.0.0:111         0.0.0.0:*            LISTEN
tcp6     0      0 :::22               :::*                  LISTEN
tcp6     0      0 :::1:25              :::*                  LISTEN
tcp6     0      0 :::443               :::*                  LISTEN
tcp6     0      0 :::5666              :::*                  LISTEN
tcp6     0      0 :::3306              :::*                  LISTEN
tcp6     0      0 :::111               :::*                  LISTEN
tcp6     0      0 :::80                :::*                  LISTEN

```

Lorsqu'on essaye de se connecter à la machine client avec le check-mk, la sortie depuis l'agent sur le port 6556 est vide :

```

bdiderot@REC160083:~$ sudo check_mk -d 172.31.131.37
Problem contacting agent: Empty output from agent at TCP port 6556

```

La machine client écoute bien sur le port 6556... Ceci est vérifié avec la commande netstat -an | grep 6556

- *Vérification de check_mk en local*

1/- ///On checke en local si le check-mk sort bien les infos ... Pour ceci, on lance le client directement dans la machine hôte et vérifiez si on récupère les données...

```

root@racktables-server:/usr/bin# check_mk_agent

```

2/- lancer le client en local avec localhost :

```

root@racktables-server:/# telnet localhost 6556

```

3/- lancer le client en local avec son fqdn :

```

root@racktables-server:/# telnet racktables-server 6556

```

- On autorise le serveur supervision à accéder au machine client pour récupérer les données. Pour ceci, il faut ajouter le serveur dans le fichier /etc/hosts.allow du machine client :

Voici le fichier /etc/hosts.allow ci-dessous :

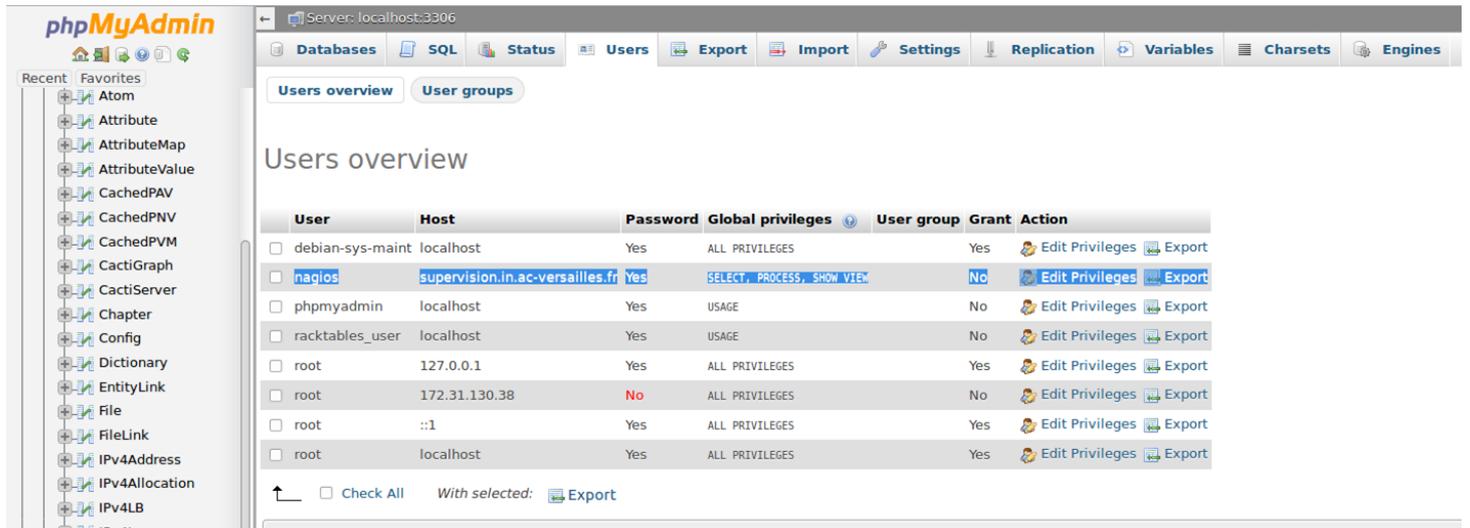
```
# /etc/hosts.allow: list of hosts that are allowed to access the system.
#       See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:  ALL: LOCAL @some_netgroup
#          ALL: .foobar.edu EXCEPT terminalserver.foobar.edu
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
ALL:172.31.131.135
##@ ip du machine supervision
```

Etat de la machine racktables-server dans la supervision après les manipules :

The screenshot shows the Nagios Check_MK interface for the host 'racktables-server'. The 'Services of Host racktables-server' section is active, displaying a table of 20 services. The services are categorized by state: OK (green), PENDING (yellow), and CRITICAL (red). The table includes columns for State, Service, Icons, Status detail, Age, Checked, and Perf-O-Meter.

State	Service	Icons	Status detail	Age	Checked	Perf-O-Meter
OK	Check_MK		OK - Agent version 1.2.6p12, execution time 0.2 sec	18 min	10 min	0.2 s
OK	Check_MK inventory		OK - no unchecked services found	13 min	8 min	
OK	[APACHE] Check active_threads		APACHE OK - active_threads 2	8 min	8 min	
OK	[APACHE] Check cpu_load		APACHE OK - cpu_load 0.001%	7 min	7 min	
OK	[APACHE] Check cpu_usage_user_children		APACHE OK - cpu_usage_user_children 0.00000%	5 min	3 min	
OK	[APACHE] Check idle_threads		APACHE OK - idle_threads 4	2 min	2 min	
PEND	[APACHE] Check requests_per_second			-	-	
OK	[APACHE] Check slots_open		APACHE OK - slots_open 145	10 min	6 min	
OK	[APACHE] Check slots_total		APACHE OK - slots_total 150	8 min	8 min	
OK	[APACHE] Check threads_closing_connection		APACHE OK - threads_closing_connection 0	8 min	8 min	
OK	[APACHE] Check threads_gracefully_finishing		APACHE OK - threads_gracefully_finishing 0	7 min	7 min	
OK	[APACHE] Check threads_idly_cleaning		APACHE OK - threads_idly_cleaning 0	5 min	3 min	
OK	[APACHE] Check threads_keepalive		APACHE OK - threads_keepalive 0	2 min	2 min	
PEND	[APACHE] Check threads_starting_up			-	-	
OK	[APACHE] Check threads_waiting_for_connection		APACHE OK - threads_waiting_for_connection 3	10 min	6 min	
OK	[APACHE] Check url		HTTP OK: HTTP/1.1 200 OK - 6481 octets en 0.003 secondes de temps de réponse	8 min	8 min	
PEND	[MYSQL] connection-time			-	-	
PEND	[MYSQL] connects-aborted			-	-	
OK	[MYSQL] log-waits		OK - 0 innodb log waits in 122 seconds (0.000/sec)	5 min	3 min	
OK	[MYSQL] long-running-procs		OK - 0 long running processes	2 min	2 min	
PEND	[MYSQL] open-files			-	-	
OK	[MYSQL] process		Uptime: 1286591 Threads: 1 Questions: 813071 Slow queries: 0 Opens: 1177 Flush tables: 1 Open tables: 199 Queries per second avg: 0.631	8 min	8 min	

// créer l'utilisateur nagios ds le mysql pour qu'il accède au bdd et attribuer tous les droits
nagios'@'supervision.in.ac-versailles.fr'



Les commandes nécessaires :

- cmk -II : permet d'inventorier les hôtes et de les configurer automatiquement
- cmk -o : ceci va créer le fichier de configuration nagios pour les hôtes et les services

option : transfert de fichier d'une machine à une autre avec la connexion ssh

```
scp check-mk-raw-1.2.8p24_0.jessie_amd64.deb root@racktables-server:/home/administrateur
```