

Mise en œuvre de Nagios

BTS SIO 2015

Samir Saidi

Sommaire

Introduction	3
I. Présentation de Nagios	4
II. Pré requis	5
<i>II.I Installation d'Apache</i>	5
<i>II.II Installation de MariaDB</i>	6
<i>II.III Installation de PHP</i>	6
<i>II.IV Installation de PHPMyAdmin</i>	8
II.V Création d'un utilisateur et groupe	9
III. Mise en œuvre de Nagios	10
III.I Téléchargements de Nagios et des Plugins	10
III.II Installation de Nagios et des Plugins	11
Installation de Nagios	11
Installation de l'interface Web de Nagios	12
Installation des plugins Nagios	12
IV. Configuration de Nagios	14
IV.I Alerte par mail	14
IV.II Accès à l'interface depuis une adresse particulière	14
IV.III Lancement de Nagios	15
V. Accès à l'interface Web	16
VI. Supervision d'un hôte Windows	17
<i>VI.I Configuration du client</i>	17
<i>VI.II Configuration du serveur</i>	18
VII. Supervision d'un hôte linux	22
<i>VII.I Configuration du client</i>	22
<i>VII.II Configuration du serveur</i>	23
VIII. Conclusion	26

Introduction

Un des rôles de l'administrateur systèmes et réseaux est de veiller au suivi et au bon fonctionnement de son infrastructure, il doit pouvoir anticiper les pannes afin d'éviter les interruptions de services. Seulement, le parc informatique d'une organisation est généralement composé de nombreux équipements, tels que des serveurs, des équipements réseaux, firewall et d'autres.

Il est assez fastidieux de se déplacer sur chaque équipement afin de vérifier son bon fonctionnement, pour cette raison il est nécessaire de mettre en œuvre une solution dite de supervision afin d'avoir une vue d'ensemble des équipements supervisés, et ceci en temps-réel.

Les outils de supervisions permettent donc de superviser plusieurs éléments physique ou services des équipements du réseau, par exemple sur les serveurs ; la mémoire disponible, les processus, les services ou encore l'espace disque restant... Par exemple un disque dur atteignant sa capacité maximal sera visible sur l'interface de la solution de supervision.

Il existe plusieurs outils de supervision, des solutions propriétaires, mais aussi des solutions open-source. Je vais dans le cadre de ce tutoriel présenter l'outil de supervision libre « Nagios » sur une distribution CentOS 7.

I. *Présentation de Nagios*

Nagios est un outil de supervision open source qui permet de surveiller l'état de divers services réseau, serveurs et autres matériels réseau.

Nagios est un programme modulaire qui se décompose en trois parties :

Le moteur de l'application : Il permet l'ordonnancement des tâches de supervision.

L'interface Web : Il s'agit du tableau de contrôle de l'application. L'interface web permet d'avoir une vue d'ensemble du système d'informations et des possibles anomalies.

Les sondes : Nagios n'est pas capable de contrôler l'état des hôtes et services sur le réseau. Pour réaliser ces opérations il fait appel à des plugins. Ces derniers sont des programmes compilés ou des scripts qui peuvent être exécutés par une ligne de commande pour contrôler l'état d'un hôte ou d'un service. Nagios va ensuite utiliser le résultat de ces plugins pour déterminer le statut des hôtes ou services sur le réseau.

II. Pré requis

II.1 Installation d'Apache

Nagios nécessite l'installation d'un serveur web Apache afin d'accéder à l'interface web. Il requiert également PHP et MySQL. Je vais alors installer ces éléments.

Installation d'Apache

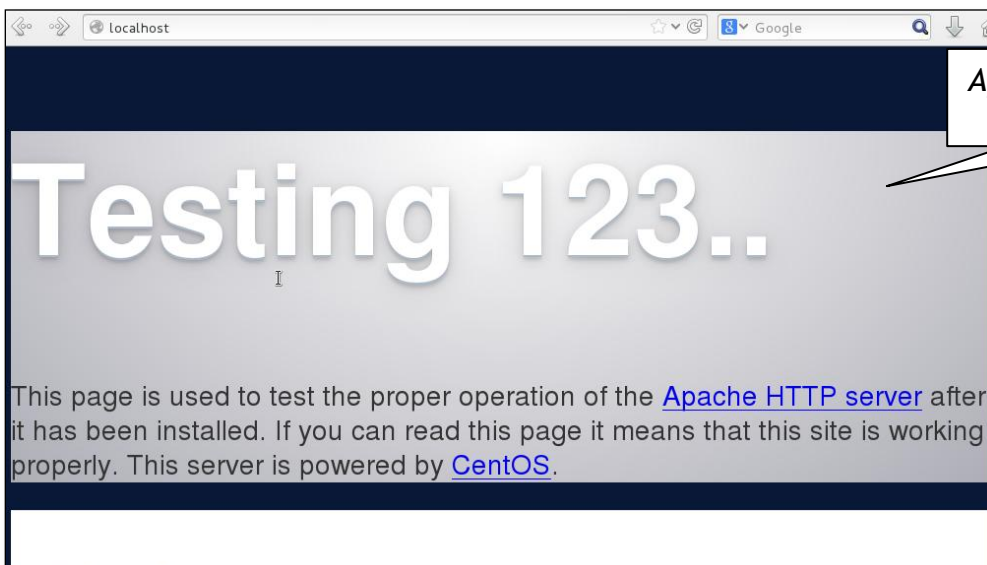
```
yum install httpd -y
```

Pour que le service Apache démarre automatiquement à chaque démarrage je saisi la commande suivante.

```
systemctl start httpd  
systemctl enable httpd
```

Je lance le service Apache, puis je saisi la commande qui permet de le démarrer automatiquement à chaque démarrage.

Je teste Apache afin en saisisant l'adresse <http://localhost/>



Apache fonctionne.

II.II Installation de MariaDB

Je vais ensuite installer MariaDB, il s'agit d'un fork de Mysql, c'est-à-dire qu'il a été développé à partir du code source de Mysql. Il se veut plus libre que Mysql.

Installation de MariaDB

```
yum install mariadb-server mariadb -y
```

Je démarre le service MariaDB et active le lancement automatique au démarrage.

```
systemctl start mariadb  
systemctl enable mariadb
```

Par défaut il n'y a pas de mot de passe pour le compte root de MySQL je vais donc configurer un mot de passe root.

```
mysql_secure_installation
```

Cette commande permet de configurer le mot de passe root.

Ensuite je saisis le nouveau mot de passe pour le compte root.
Pour chaque requête suivante je choisis « YES »

II.III Installation de PHP

J'installe PHP avec la commande suivante :

```
yum install php php-mysql php-gd php-pear -y
```

Pour tester PHP, je vais créer un fichier test :

```
vi /var/www/html/testphp.php
```

La page testphp.php est créée.

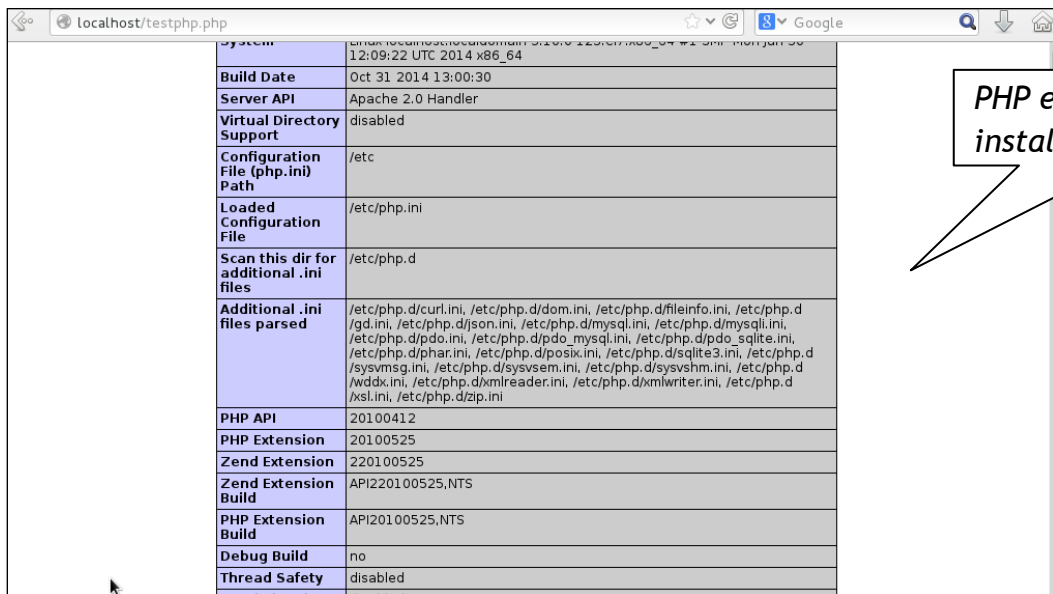
avec les lignes suivantes

```
<?php  
phpinfo();  
?>
```

Je redémarre le service httpd :

```
systemctl restart httpd
```

Je vérifie que PHP fonctionne, en saisissant <http://localhost/test.php.php> sur le navigateur :



System	Linux localhost.localdomain 3.10.0-123.el7.x86_64 #1 SMP Mon Jun 30 12:09:22 UTC 2014 x86_64
Build Date	Oct 31 2014 13:00:30
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/dom.ini, /etc/php.d/fileinfo.ini, /etc/php.d/gd.ini, /etc/php.d/json.ini, /etc/php.d/mysql.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/posix.ini, /etc/php.d/sqlite3.ini, /etc/php.d/sysvmsg.ini, /etc/php.d/sysvsem.ini, /etc/php.d/sysvshm.ini, /etc/php.d/wddx.ini, /etc/php.d/xmlreader.ini, /etc/php.d/xmlwriter.ini, /etc/php.d/xsl.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API220100525,NTS
PHP Extension Build	API20100525,NTS
Debug Build	no
Thread Safety	disabled

PHP est correctement installé

II.IV Installation de PHPMyAdmin

Pour gérer les bases de données MariaDB je vais installer PHPMyAdmin

Le dépôt EPEL doit être installé.

```
yum install phpmyadmin -y
```

Par défaut PHPMyAdmin est seulement accessible en local, pour le rendre accessible de partout je vais modifier le fichier phpmyadmin.conf

```
vi /etc/httpd/conf.d/phpMyAdmin.conf
```

et commenter entièrement la section /<Directory> puis ajouter des lignes

```
[...]
Alias /phpMyAdmin /usr/share/phpMyAdmin
Alias /phpmyadmin /usr/share/phpMyAdmin

    Je commentes la section suivante :

#<Directory /usr/share/phpMyAdmin/>
# <IfModule mod_authz_core.c>
#   # Apache 2.4
#   <RequireAny>
#     Require ip 127.0.0.1
#     Require ip ::1
#   </RequireAny>
# </IfModule>
# <IfModule !mod_authz_core.c>
#   # Apache 2.2
#   Order Deny,Allow
#   Deny from All
#   Allow from 127.0.0.1
#   Allow from ::1
# </IfModule>
#</Directory>

    J'ajoute les lignes suivantes :

<Directory /usr/share/phpMyAdmin/>
    Options none
    AllowOverride Limit
    Require all granted
</Directory>
[...]
```


Pour changer l'authentification sur PHPMyAdmin je vais modifier le fichier config.inc.php et changer « cookie » en « http »

```
vi /etc/phpMyAdmin/config.inc.php
```

```
[...]
```

```
/* Authentication type */
```

```
$cfg['Servers'][$i]['auth_type'] = 'http'; // Authentication  
method (config, http or cookie based)?
```

L'authentification par http est sélectionnée.

```
[...]
```

Je redémarre le service Apache

```
systemctl restart httpd
```

Un dernier pré requis est nécessaire

```
yum install gd gd-devel gcc glibc glibc-common wget
```

II.V Création d'un utilisateur et groupe

Je vais créer un nouvel utilisateur et lui donner un mot de passe

```
useradd -m nagios
```

```
passwd nagios
```

Je créer ensuite un groupe :

```
groupadd nagcmd  
usermod -a -G nagcmd nagios  
usermod -a -G nagcmd apache
```

Je viens d'installer les pré-requis nécessaires, je vais maintenant pouvoir passer à l'installation de Nagios.

III. Mise en œuvre de Nagios

III.1 Téléchargements de Nagios et des Plugins

Pour télécharger Nagios

Téléchargement de Nagios

```
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
```

Puis les plugins

Téléchargement des plugins

```
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
```

III.II *Installation de Nagios et des Plugins*

Installation de Nagios

Je vais maintenant procéder à l'installation de Nagios, pour cela je vais aller à l'emplacement où Nagios a été téléchargé puis je vais extraire le contenu de Nagios.

```
tar xzf nagios-4.0.8.tar.gz
```

Ensuite je vais me déplacer sur le répertoire de nagios avec la commande :

```
cd nagios-4.0.8
```

et je vais exécuter une série de commande afin de compiler et d'installer Nagios.

```
./configure --with-command-group=nagcmd  
make all  
make install  
make install-init  
make install-config  
make install-commandmode
```

Installation de l'interface Web de Nagios

```
make install-webconf
```

Je vais créer un compte pour me connecter par la suite à l'interface web de Nagios puis choisir un mot de passe.

Le compte créer est « nagiosadmin »

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Je redémarre le service Apache

```
systemctl restart httpd
```

Installation des plugins Nagios

Dans le répertoire qui contient les plugins téléchargés précédemment je saisis la commande qui permet de décompresser l'archive :

```
tar xzf nagios-plugins-2.0.3.tar.gz
```

Je me déplace dans le dossier des plugins

```
cd nagios-plugins-2.0.3
```

et je compile et installe :

```
./configure --with-nagios-user=nagios --with-nagios-group=nagios  
make  
make install
```

L'installation de Nagios est maintenant terminée mais il reste maintenant à le configurer.

IV. Configuration de Nagios

Les fichiers de configuration de Nagios ont été installés dans `/usr/local/nagios/etc`. Je vais me servir des fichiers contenu dans ce répertoire afin de modifier certains paramètres.

IV.I Alerte par mail

En cas de notification il est possible d'être alerté par email, je vais donc saisir mon adresse mail dans le fichier suivant :

```
vi /usr/local/nagios/etc/objects/contacts.cfg
```

```
define contact{
    contact_name      nagiosadmin      ; Short name of user
    use                generic-contact  ; Inherit default values from generic-contact te
    template (defined above)
    alias              Nagios Admin    ; Full name of user
    email              nagios@localhost ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>>
}
**
```

IV.II Accès à l'interface depuis une adresse particulière

Pour accéder à l'interface Web depuis une adresse autre que local, il faut modifier le fichier suivant :

```
vi /etc/httpd/conf.d/nagios.conf
```

```
[...]
## Je commente ces lignes  ##
# Order allow,deny
# Allow from all

## Je décommente et ajoute l'adresse  ##
Order deny,allow
Deny from all
Allow from 127.0.0.1 192.168.1.0/24
[...]
```

Un accès administrateur sera donc possible depuis les adresses en 192.168.1.0/24

Je redémarre le service Apache.

```
systemctl restart httpd
```

IV.III Lancement de Nagios

La configuration et l'installation est maintenant effectué, je vais donc lancer Nagios, mais avant tout je vérifie si il n'y a pas d'erreur de configuration :

```
/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

Ensuite je lance le service Nagios puis je saisis la commande qui permet de le démarrer automatiquement à chaque démarrage.


```
systemctl start nagios
chkconfig --add nagios
chkconfig nagios on
```

Je vais pouvoir accéder à l'interface Web de Nagios.

V. Accès à l'interface Web

L'adresse pour accéder à l'interface Web de Nagios est <http://127.0.0.1/nagios>

Authentification requise

 Le site <http://127.0.0.1> demande un nom d'utilisateur et un mot de passe. Le site indique : « Nagios Access »

Utilisateur :

Mot de passe :

Le mot de passe correspond à celui choisi dans les étapes précédentes et l'utilisateur est « nagiosadmin »

Voici l'interface Web de Nagios :

The screenshot shows the Nagios Core web interface. At the top, the Nagios Core logo is displayed with a green checkmark and the text "Daemon running with PID 9358". Below this, the version "Nagios Core Version 4.0.8" is shown, along with the date "August 12, 2014" and a "Check for updates" link. The interface is divided into several sections: "General" with links for Home and Documentation; "Current Status" with links for Tactical Overview, Map, Hosts, Services, Host Groups, and Service Groups; "Problems" with links for Services (Unhandled), Hosts (Unhandled), and Network Outages; "Reports" with links for Availability, Trends, and Alerts; "Get Started" with a list of links for starting monitoring, changing the look, extending Nagios, getting support, training, and certification; "Quick Links" with links to Nagios Library, Labs, Exchange, Support, .com, and .org; "Latest News" with links for NCPA 1.7.2 and 1.7.1 releases, and Nagios Core 4.0.8 release; and "Don't Miss..." with links for Nagios Log Server Dashboard Contest, Nagios Log Server webinar, and Nagios World Conference. A video player is visible on the right side of the dashboard.

Pour le moment seulement le serveur Nagios est superviser, il faut donc ajouter des hôtes c'est ce que je vais faire dans les étapes suivante.

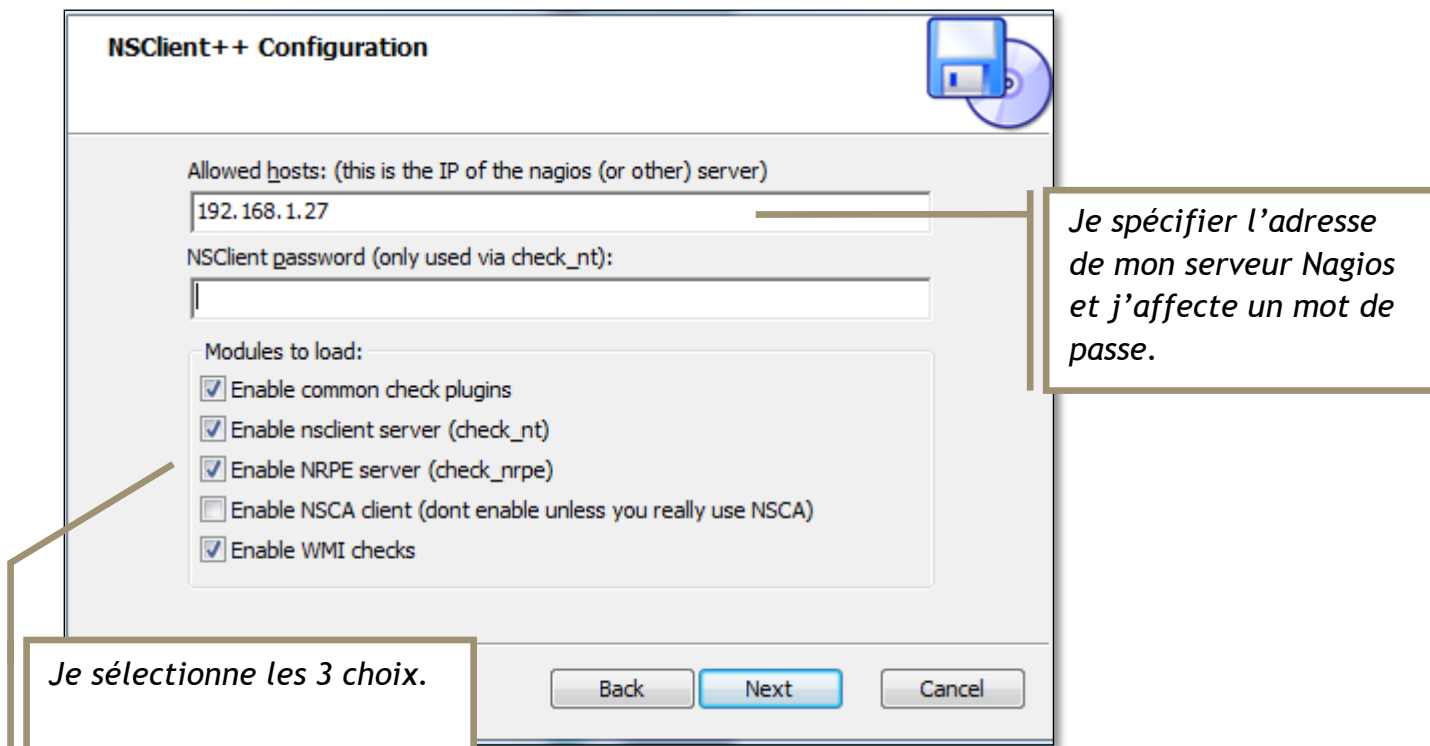
VI. Supervision d'un hôte Windows

VI.I Configuration du client

Afin de superviser un hôte utilisant Windows, il faut utiliser le service NSClient ++.

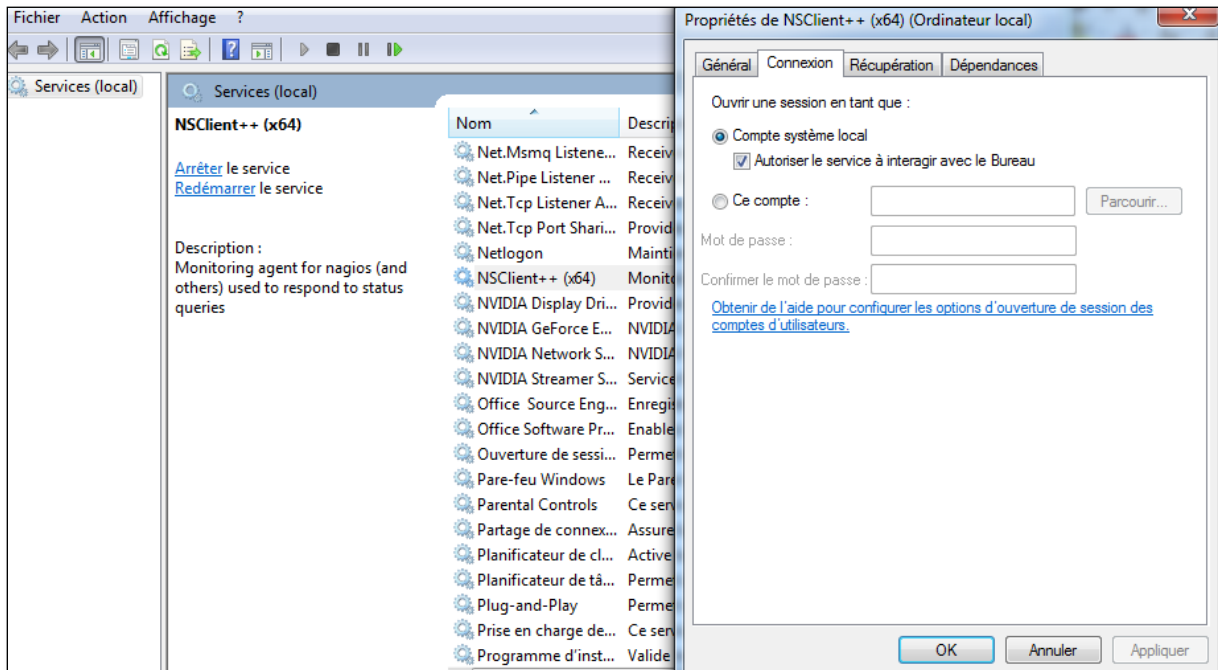
[NSClient++](#)

Une fois le telechargement terminé j'execute le fichier MSI et je lance l'installation.



A la fin de l'installation il faut laisser cocher « Démarrer le service »

Il faut maintenant autoriser le service NSClient à interagir avec le bureau, pour cela je vais dans services.msc et je coche la case correspondante.



VI.II Configuration du serveur

Je vais maintenant me rendre sur mon serveur Nagios afin de le configurer pour qu'il puisse superviser un hôte Windows.

Dans un premier il faut définir la définition d'objet windows.cfg dans le fichier de configuration nagios.cfg en décommentant la ligne correspondante :

```
# vi /usr/local/nagios/etc/nagios.cfg  
  
cfg_file=/usr/local/nagios/etc/objects/windows.cfg
```

Je décommente cette ligne

Ensuite il faut configurer la protection mot de passe de NSClient++

```
# vi /usr/local/nagios/etc/objects/commands.cfg
```

Dans ce fichier il faut ajouter l'argument « -s » suivi du password choisis lors de l'installation de NSClient++.

```
define command{  
  
    command_name    check_nt  
  
    command_line    $USER1$/check_nt -H $HOSTADDRESS$ -p 12489 -s  
yourpassword -v $ARG1$ $ARG2$  
  
}
```

Ensuite il faut éditer le fichier de configuration windows.cfg

```
# vi /usr/local/nagios/etc/objects/windows.cfg
```

Il faut remplacer les valeurs “host_name”, “alias” et “address” avec les valeurs du serveur Nagios.

Le champ « Host_name » définit pour la section « define host » doit correspondre à celui définit dans la section « define_service ».

```
define host{

    use                windows-server ; Inherit default values from a
Windows server template (make sure you keep this line!)

    host_name          Windows 7

    alias              Seven

    address            192.168.1.32

}

define service{

    use                generic-service

    host_name          Windows 7

    service_description NSClient++ Version

    check_command      check_nt!CLIENTVERSION

}
```

Une fois les configurations effectuées je vérifie qu'il n'y a pas d'erreurs en lançant la commande suivante :

```
# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

Ensuite je retourner sur l'interface web de Nagios pour vérifier si l'hôte est bien ajouter :

The screenshot shows the Nagios web interface for a host named 'Windows Seven (Windows 7)' with IP address 192.168.1.32. The host status is 'UP' (for 0d 7h 49m 27s). The interface includes sections for Host Information, Host State Information, and Host Commands. A callout box points to the IP address, stating: 'Mon hôte Windows 7 d'adresse 192.168.1.32 est bien reconnu par Nagios'.

Host State Information	
Host Status:	UP (for 0d 7h 49m 27s)
Status Information:	PING OK - Paquets perdus = 0%, RTA = 4.80 ms
Performance Data:	rta=4.795000ms;3000.000000;5000.000000; 0.000000 pl=0%;80;100;0 1/10 (HARD state)
Current Attempt:	ACTIVE
Last Check Time:	02-13-2015 23:01:44
Check Type:	ACTIVE
Check Latency / Duration:	0,000 / 4,000 seconds
Next Scheduled Active Check:	02-13-2015 23:06:49
Last State Change:	02-13-2015 15:12:46
Last Notification:	N/A (notification 0)
Is This Host Flapping?	NO (0,00% state change)
In Scheduled Downtime?	NO
Last Update:	02-13-2015 23:02:11 (0d 0h 0m 2s ago)
Active Checks:	ENABLED
Passive Checks:	ENABLED
Obsessing:	ENABLED
Notifications:	ENABLED
Event Handler:	ENABLED
Flap Detection:	ENABLED

Le client Windows 7 est ajouté plusieurs services peuvent ainsi êtres surveillés comme la capacité restante du disque dur ou encore la charge CPU...

VII. Supervision d'un hôte linux

VII.I Configuration du client

Sur le client linux il faut installer « NRPE » ainsi que « nagios plugins ».

NRPE est un « Addons » pour Nagios qui permet d'exécuter des greffons (plugins) sur un serveur (Linux/Unix ou Windows) distant.

```
yum install nrpe nagios-plugins-all openssl
```

Je vais maintenant ajouter l'adresse IP du serveur Nagios en éditant le fichier nrpe.scg :

```
sudo vi /etc/nagios/nrpe.cfg
```

```
[...]
##      Il faut rajouter l'adresse IP du serveur Nagios      ##
allowed_hosts=127.0.0.1 192.168.1.150
[...]
```

Je démarre le service NRPE :

```
systemctl start nrpe
chkconfig nrpe on
```

VII.II Configuration du serveur

Il faut maintenant ajouter les clients dans le fichier de configuration du serveur :

Pour cela il faut éditer le fichier : “/usr/local/nagios/etc/nagios.cfg”

```
vi /usr/local/nagios/etc/nagios.cfg
```

et décommenter la ligne suivante :

```
cfg_dir=/usr/local/nagios/etc/servers
```

Ensuite je vais créer un répertoire appelé « serveurs » sur le chemin “/usr/local/nagios/etc/”.

```
mkdir /usr/local/nagios/etc/serveurs
```

Puis je créer un fichier de configuration pour les clients supervisés.

```
vi /usr/local/nagios/etc/serveurs/clients.cfg
```

et j'ajoute les lignes suivantes :

```
define host{  
  
    use                linux-server  
  
    host_name          client  
  
    alias              client  
  
    address            192.168.1.152  
  
    max_check_attempts 5  
  
    check_period       24x7  
  
    notification_interval 30  
  
    notification_period 24x7  
  
}
```

*Il s'agit de l'adresse
du client Nagios*

Enfin je redémarre le service Nagios

```
systemctl restart nagios
```

Dans le cas où plusieurs clients doivent être supervisés, il faut simplement créer un fichier de configuration séparé pour chaque client dans le répertoire «/usr/local/nagios/etc/serveurs».

La configuration du serveur est maintenant terminée, l'hôte est donc ajouté sur l'interface de Nagios.

VIII. Conclusion

Je viens donc d'expliquer comment mettre en place une solution de supervision pour une architecture informatique, comme je viens de le constater il est très pratique de disposer de cet outils de supervision car lorsque plusieurs serveurs doivent être supervisé, cela est beaucoup plus simple à gérer quand ceux-ci sont centralisé au sein d'une même interface comme le propose Nagios.