TP Nagios : supervision des réseaux

Objectif : Appréhender les notions de supervision des réseaux. Utilisation et déploiement du logiciel Nagios.

Nagios (anciennement appelé Netsaint) est une application permettant la surveillance système et réseau. Elle surveille les hôtes et services spécifiés, alertant lorsque les systèmes vont mal et quand ils vont mieux. C'est un logiciel libre sous licence GPL.

C'est un programme modulaire qui se décompose en trois parties :

1. Le moteur de l'application qui vient ordonnancer les tâches de supervision.
2. L'interface web, qui permet d'avoir une vue d'ensemble du système d'information et des possibles anomalies.
3. Les plugins, une centaine de mini programmes que l'on peut compléter en fonction des besoins de chacun pour superviser chaque service ou ressource disponible sur l'ensemble des ordinateurs ou éléments réseaux du SI.

1 : Installation de Nagios à partir d'un dépôt distant (Nagios 2.9)

Adresse du dépôt :
http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/10.3/repo/oss/suse/x86_64/

•Le gestionnaire de paquets : Package manager : zypper

http://en.opensuse.org/Zypper

Cas d'utilisations :

```
zypper                # to print the list of available global options and commands
zypper help search    # to print help for the search command
zypper lp             # to see what patch updates are needed
zypper patch          # to apply the needed patches
zypper se sqlite      # to search for sqlite
zypper rm sqlite2     # to remove sqlite2
zypper in sqlite3     # to install sqlite3
zypper in yast*       # to install all packages matching 'yast*'
zypper up             # to update all installed packages with newer versions, where possible
```

•1.1 Ajout d'un dépôt sous ZYPPER :

Syntaxe : zypper ar <URL> alias
Dans notre cas :
zypper ar http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/10.3/repo/oss/suse/ nagios

ça ne marche pas !!!!!! mais c'est normal, la connexion HTTP nécessite une authentification. Zypper est fondé sur « curl » pour le rapatriement HTTP, ce dernier doit être paramétré pour franchir le proxy HTTP.

Alors on modifie le fichier /root/.curl ou /root/.curlrc avec « vi » par exemple.

/root/.curlrc
 # Changed by YaST2 module proxy 19.11.2007
--proxy-user = "login:motdepasse"
--proxy "http://wwwcache.univ-lr.fr:3128"

Ensuite on refait :
zypper ar http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/10.3/repo/oss/suse/ nagios

zypper ar http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/10.3/repo/oss/suse/ nagios
** Adding repository 'nagios'*
Repository 'nagios' successfully added:
Enabled: Yes
Autorefresh: Yes
URL: http://ftp5.gwdg.de/pub/opensuse/discontinued/distribution/10.3/repo/oss/suse/

On vérifie que le dépôt est ajouté : ***zypper lr***

```
archi053:~ # zypper lr
# | Enabled | Refresh | Type   | Alias                 | Name
--+---------+---------+--------+-----------------------+----------------------
1 | Yes     | No      | yast2  | openSUSE-10.3-DVD 10.3 | openSUSE-10.3-DVD 10.3
2 | Yes     | Yes     | rpm-md | nagios                | nagios
archi053:~ #
```

On s'aperçoit qu'il y a un autre dépôt openSUSE-10.3-DVD 10.3. On le supprime car nous n'avons pas le CD d'installation.

Suppression du dépôt existant : **zypper rr "openSUSE-10.3-DVD 10.3"**

•1.2 Installation de Nagios avec ZYPPER

Taper la ligne suivante :
zypper install nagios nagios-www nagios-plugins nagios-plugins-extras nagios-nrpe rrdtool php5 php5-gd php5-zlib apache2-mod_php5 perl-SNMP net-snmp-32bit nmap ncpfs libwavpack1

Nagios s'appuie sur plusieurs briques logiciels. APACHE, PHP, PERL, ….

•Démarrage des services

chkconfig apache2 on
chkconfig nagios on

service apache2 start
service nagios start

Vérification de l'exécution des daemons.... plein de manières de faire
- ps -ef|grep nagios
-. /etc/init.d/nagios status
- service nagios status

1.3 Utilisation de l'interface web. IHM légère : Front-end Web

Créer un utilisateur web pour l'authentification HTTP. Apache fournit l'outil htpasswd2 pour gérer les utilisateurs et leur mot de passe. L'accès à l'IHM Nagios (http://localhost/nagios) est protégé par une authentification (.htaccess).

Créer l'utilisateur : nagiosadmin

htpasswd2 -c /etc/nagios/htpasswd.users nagiosadmin

Éditer le fichier **/etc/apache2/cond.d/nagios.conf**

Modifier le pour qu'il soit conforme à l'exemple ci-dessous:

```
ScriptAlias /nagios/cgi-bin /usr/lib/nagios/cgi
<Directory /usr/lib/nagios/cgi>
   Options ExecCGI
   order deny,allow
   Allow from all
   AuthName "Nagios Access"
   AuthType Basic
   AuthUserFile /etc/nagios/htpasswd.users
   Require valid-user
</Directory>

Alias /nagios  /usr/share/nagios
<Directory /usr/share/nagios>
   Options None
   order deny,allow
   Allow from all
    AuthName "Nagios Access"
    AuthType Basic
    AuthUserFile /etc/nagios/htpasswd.users
    Require valid-user
</Directory>
```

Redémarrage des services :
service apache2 restart
service nagios restart

Tester en vous authentifiant : http://localhost/nagios

Regarder soigneusement la documentation html.

2. Configuration de Nagios

2.1 Où les fichiers de Nagios sont ils installés ?

2.2 Quels sont les fichiers de configuration de Nagios ?

2.3 Vérification que les fichiers de configuration sont valides :

/usr/sbin/nagios -v nagios.cfg

2.4 Regarder le fichier nagios.cfg et désactivé l'emploi du fichier localhost.cfg

2.5 Regarder le fichier nagios.cfg et activé l'emploi des commandes externes

2.6 Regarder le fichier nagios.cfg et imposé un intervalle de temps de 15s pour la vérification des commandes externes :

2.7 Regarder le fichier cgi.cfg et activer l'authentification pour l'utilisateur nagiosadmin.

2.8 Analyser le fichier localhost.cfg

2.9 A quoi sert le fichier commands.cfg ?

2.10 Où sont placer les plugins de Nagios, c'est à dire, ses outils de vérification ?

2.11 Donner un exemple d'utilisation du plugin : check_ping (check_ping -h)

3. Ajouté une machine à monitorer (Machine noire sous windows)

3.1 Mise en place d'un équipement

3.1.1 Créer une période de temps :

Tout d'abord, il faut définir la période d'utilisation. Créer un fichier **timeperiods.cfg.** Dans **localhost.cfg** vous avez ces périodes et choisissez celle que vous voulez utiliser 24x7 dans notre cas)

 # '24x7' timeperiod definition
define timeperiod{
timeperiod_name 24x7
alias 24 Hours A Day, 7 Days A Week
sunday 00:00-24:00
monday 00:00-24:00
tuesday 00:00-24:00
wednesday 00:00-24:00
thursday 00:00-24:00
friday 00:00-24:00
saturday 00:00-24:00
}

3.1.2 Créer un contact

Il faut ensuite créer un contact dans **contacts.cfg**. Il y a par défaut le contact **nagios (localhost.cfg)**, mais vous pouvez vous en créer un autre si vous voulez.

 # 'nagios' contact definition
define contact{
contact_name nagios
alias Nagios Admin
service_notification_period 24x7
host_notification_period 24x7
service_notification_options w,u,c,r
host_notification_options d,u,r
service_notification_commands notify-by-email,notify-by-epager
host_notification_commands host-notify-by-email,host-notify-by-epager
email nagios-admin@localhost
pager pagenagios-admin@localhost
}

3.1.3 Créer un groupe de contacts
Il faut ensuite créer dans le fichier **contactgroups.cfg** un groupe de contacts**.**

# 'serveur-admin' contact group definition
define contactgroup{
contactgroup_name serveur-admin
alias Serveurs
members nagios
}

3.1.4 Créer une machine

Maintenant il faut créer la machine. Créer la dans **hosts.cfg**. Créer d'abord un équipement générique en vous appuyant sur le fichier **localhost.cfg** . Puis définissez votre propre machine : En guise d'exemple ? Voici un exemple caduc avec une machine dont l'adresse est 10.10.10.10

 # Serveur nagios host definition
define host{
use generic-host ; Name of host template to use

host_name serveur
alias Serveur
address 10.10.10.10
check_command check-host-alive
max_check_attempts 23
contact_groups serveur-admin
notification_interval 60
notification_period 24x7
notification_options d,u,r
}

Dans **hostgroups.cfg**, créer le groupe de votre équipement auquel il appartiendra. En exemple :

# Serveur host group definition
define hostgroup{
hostgroup_name serveurs
alias Serveurs
members serveur
}

4. Ajouter un service à vérifier :

Et finalement créer le fichier **services.cfg**. Créer un service générique en vous inspirant du fichier **localhost.cfg**. Ensuite, définissez le service **ping**.

define service{
use generic-service ; Name of service template to use

host_name serveur
service_description PING
is_volatile 0
check_period 24x7
max_check_attempts 3
normal_check_interval 5
retry_check_interval 1
contact_groups serveur-admin
notification_interval 20
notification_period 24x7
notification_options c,r
check_command check_ping
}

-Ensuite, il suffit juste maintenant d'aller dans **nagios.cfg** et de décocher les lignes :

cfg_file=/usr/local/nagios/etc/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/contacts.cfg
cfg_file=/usr/local/nagios/etc/contactgroups.cfg
cfg_file=/usr/local/nagios/etc/hosts.cfg
cfg_file=/usr/local/nagios/etc/hostgroups.cfg
cfg_file=/usr/local/nagios/etc/services.cfg

- Vérification des erreurs dans les fichiers de configuration :
Si il y a un quelconque problème vous pouvez voir l'erreur avec la commande :

/usr/local/nagios/bin/nagios –v /usr/local/nagios/etc/nagios.cfg

- Relancer nagios :
/etc/init.d/nagios restart

Votre équipement est maintenant à l'état « pending » pendant une dizaine de minute et se mettra après à l'état UP par la suite. *http://localhost/nagios*

5. Les tests

5.1 Effectuer les tests qui vous semblent pertinent pour montrer le bon fonctionnement de Nagios. Montrer que Nagios suit bien l'activité de votre machine sous windows ? (les impressions d'écrans sont les bienvenues)

6. Ajouter les informations adéquates permettant de vérifier le bon fonctionnement du service DNS sur votre machine Linux (Le serveur Nagios). Le test DNS consistera à retourner l'adresse IP de la machine wwwcache.univ-lr.fr en utilisant le serveur DNS suivant : 10.2.40.231 (mahonet)