

Supervision



- 1 Nagios
- 2 Nagios et snmp

Nagios

- 1 Nagios
- 2 Nagios et snmp

Introduction

Célèbre logiciel de supervision

- ▶ libre (Nagios Core)
- ▶ performant
- ▶ populaire
- ▶ modulaire (plugins)

Architecture

- ▶ monitoring
- ▶ alerte
- ▶ réponse (confirmer, escalade, ...)
- ▶ maintenance (interruption de service planifiée)
- ▶ analyse (visualisation des tendances, rapports de disponibilité, ...)

États possibles

Une machine peut être :

- ▶ OK
- ▶ PENDING
- ▶ DOWN
- ▶ UNREACHABLE

Un service peut être :

- ▶ OK
- ▶ WARNING
- ▶ UNKNOWN
- ▶ PENDING
- ▶ CRITICAL

Références

- ▶ Site principal
- ▶ Manuel (v4)
- ▶ Définition des objets

Architecture

- ▶ 1 service chargé d'ordonner et d'exécuter les différents tests en place, récupérer leur résultat et alerter au besoin.
- ▶ 1 interface web

Installation

Nécessite le dépôt *EPEL* pour ensuite installer le paquet *nagios*

Interface web

Est définie par le fichier */etc/httpd/conf.d/nagios.conf*

Notez l'utilisation de l'authentification http (*/etc/nagios/passwd*)

Exercice

- ▶ Installer nagios
- ▶ Activer et lancer le service
- ▶ Configurer l'interface web

Cette interface web doit être accessible

Introduction

Toutes les difficultés d'une supervision efficace sont résumées dans les questions suivantes :

- 1 Que surveiller ?
- 2 Comment ?

Question : déterminer ce que l'on souhaite surveiller

La configuration nagios

Consiste à

- 1 Définir les objets (services, commandes ,groupe de machines, machines)
- 2 Définir leurs relations
- 3 Tout en conservant une configuration maintenable

Validez !

Pensez à vérifier et valider les modifications apportées à la conf avant de relancer le service :

```
nagios -v /etc/nagios/nagios.cfg
```

```
Nagios Core 3.2.1
```

```
Copyright (c) 2009-2010 Nagios Core Development Team and Community Contributors
```

```
Copyright (c) 1999-2009 Ethan Galstad
```

```
Last Modified: 03-09-2010
```

```
License: GPL
```

```
Website: http://www.nagios.org
```

```
Reading configuration data...
```

```
  Read main config file okay...
```

```
Processing object config file '/etc/nagios/commands.cfg'...
```

```
Processing object config directory '/etc/nagios-plugins/config'...
```

```
Processing object config file '/etc/nagios-plugins/config/ntp.cfg'...
```

```
Processing object config file '/etc/nagios-plugins/config/nt.cfg'...
```

```
...
```

```
Checking for circular host and service dependencies...
```

```
Checking global event handlers...
```

```
Checking obsessive compulsive processor commands...
```

```
Checking misc settings...
```

```
Total Warnings: 1
```

```
Total Errors: 0
```

```
Things look okay - No serious problems were detected during the pre-flight check
```

If all else fails

Pensez à jeter un oeil dans `/var/log/nagios/nagios.log`

Les fichiers de configuration

Dans `/etc/nagios`

`cgi.cfg` configuration de l'interface web, apparence et contrôle d'accès

`nagios.cfg` configuration principale

`nrpe.cfg` configuration du client *nrpe*

`objects/*.cfg` définition de différents objets

Exercice : modifiez la configuration de nagios afin qu'il prenne en compte vos (futurs) fichiers de configuration que vous stockerez dans `/etc/nagios/conf.d`. Déplacez les fichiers de `/etc/nagios/objects` vers ce répertoire.

Factorisez !

BAD !

```
define host{
  name      caffeine.epsi.fr
  notifications_enabled 1
  event_handler_enabled 1
  flap_detection_enabled 1
  failure_prediction_enabled 1
  process_perf_data 1
  retain_status_information 1
  retain_nonstatus_information 1
  notification_period 24x7
  icon_image base/debian.png
  address caffeine
}
```

```
define host{
  name      workine.epsi.fr
  notifications_enabled 1
  event_handler_enabled 1
  flap_detection_enabled 1
  failure_prediction_enabled 1
  process_perf_data 1
  retain_status_information 1
  retain_nonstatus_information 1
  notification_period 24x7
  icon_image base/redhat.png
  address workine
}
```

Factorisez !

GOOD!

```
define host{
  name      generic-host
  notifications_enabled 1
  event_handler_enabled 1
  flap_detection_enabled 1
  failure_prediction_enabled 1
  process_perf_data 1
  retain_status_information 1
  retain_nonstatus_information 1
  notification_period 24x7
  register 0
}
```

```
define host{
  name      workine.epsi.fr
  use      generic-host
  icon_image base/redhat.png
  address   workine
}
```

```
define host{
  name      cafeine.epsi.fr
  use      generic-host
  icon_image base/debian.png
  address   cafeine
}
```

Les machines

host definition

Les principales directives :

host_name

alias

address

max_check_attempts

check_period 24x7

contacts

contact_groups

notification_interval

notification_period 24x7

Contrôle des notifications

`check_command` commande utilisée pour vérifier la *connectivité* de l'équipement

`max_check_attempts` ≥ 1

`check_interval`

`retry_interval`

Question : comment ne pas recevoir de notification le weekend ?

Les groupes de machines

hostgroup definition

Principalement utilisé pour éviter les redondances dans la surveillance des services.

Les services

Un objet *service* associe une *machine*, ou (mieux) un *groupe de machines* à une *commande*

```
define service{
    use generic-service ; Name of service template
    to use
    hostgroup_name toutlemonde
    service_description PING
    check_command check_ping!100.0,20%!500.0,60%
}
```

Les services

service definition

Les services associent machine(s) et services (à surveiller)

host_name

host_description

check_command

max_check_attempts

check_interval

retry_interval

check_period

notification_interval

notification_period

contacts

contact_groups

Question : comment éviter la répétition d'une notification ?

Les groupes de services

Utilisés afin de simplifier l'administration et de personnaliser les affichages sur l'interface

Les plugins

- ▶ S'installent via les paquets *nagios-plugins-**
- ▶ Servent à réaliser différents tests.
- ▶ Sont dans */usr/lib64/nagios/plugins*
- ▶ Peuvent être utiliser directement

```
/usr/lib64/nagios/plugins/check_imap -H courrier.opendoor.  
fr
```

Plugins nagios : options communes

- ▶ `/usr/lib64/nagios/plugins/check_NAME --help` pour avoir le détail du plugin.
- ▶ `/usr/lib64/nagios/plugins/check_NAME --host` ou `-H` pour spécifier le nom de machine à tester.

Les commandes

Les commandes nagios sont construites autour d'un plugin, avec des options prédéfinies, et d'autres paramétrables

```
define command{  
    command_name      check_ping  
    command_line      $USER1$/check_ping -H $HOSTADDRESS$ -w  
                      $ARG1$ -c $ARG2$ -p 5  
}
```

Les éléments entre \$\$ sont des *macros*

Au boulot

Identifier et utiliser le plugin permettant de vérifier l'état du serveur web *http://opendoor.fr* et qui renvoie :

- ▶ **CRITICAL** si le site ne répond pas
- ▶ **CRITICAL** s'il met plus de 1sec à répondre
- ▶ **OK** s'il met moins de 1sec à répondre
- ▶ **WARNING** dans les autres cas

Mettez en place cette surveillance via nagios

Nagios et snmp

- 1 Nagios
- 2 Nagios et snmp

Exercice

Utiliser nagios et son plugin `check_snmp` afin de surveiller les éléments critiques déterminés précédemment.

On s'aidera notamment des scripts disponibles [ici](#)