IRIS - ECOLE SUPERIEURE
D'INFORMATIQUE

Nagios®

LA SUPERVISION RÉSEAU ET SYSTÈME AVEC NAGIOS CORE SOUS UBUNTU SERVER (SNMP)

Ajay MUTHU KUMAR
Alan TOTH-SIZAIRE

BTS SIO LM 1

SOMMAIRE

DEFINITION DE LA SUPERVISION ——— P1 - P2

ETAPE 1 : INSTALLATION DE LA MACHINE UBUNTU ——— P3 - P8

ETAPE 2 : CONFIGURATION DE LA MACHINE UBUNTU ——— P9 - P12

ETAPE 3 : INSTALLATION DE NAGIOS CORE ——— P13 - P19

ETAPE 4 : INSTALLATION DES PLUGINS NAGIOS ——— P20 - P22

ETAPE 5 : SUPERVISION DES HÔTES ——— P23 - P27

ETAPE 6 : CONFIGURATION DES NOTIFICATIONS PAR MAIL – P28 - P30

EXPLICATIONS ——— P31 - P31

ETAPE FINAL : VERIFICATION DE LA RECEPTION DES NOTIFICATIONS ——— P32 - P34

Définition de la Supervision

La supervision désigne l'ensemble des processus, outils et méthodes permettant de surveiller un système informatique en temps réel afin de détecter des anomalies, des dysfonctionnements ou des menaces potentielles. Elle inclut la collecte, l'analyse et la corrélation d'événements issus de divers composants d'un réseau, comme les serveurs, les bases de données, les applications ou les équipements de sécurité (pare-feu, IDS/IPS, etc.).

Elle peut être classée en plusieurs catégories :

- Supervision technique : surveille l'état des infrastructures (CPU, RAM, stockage, disponibilité des services, etc.).
- Supervision applicative : suit le bon fonctionnement des logiciels métiers.
- Supervision de sécurité : détecte et analyse les menaces et incidents de sécurité via des solutions comme un SIEM (Security Information and Event Management).

Importance de la supervision dans un audit de sécurité

Dans le cadre d'un audit de sécurité, la supervision est essentielle pour plusieurs raisons :

1. Détection précoce des incidents : Une bonne supervision permet d'identifier rapidement des comportements anormaux (tentatives d'intrusion, activités suspectes, erreurs système critiques).

2. Traçabilité et journalisation : Les logs collectés et corrélés servent de base pour l'analyse forensic en cas d'incident.

3. Conformité réglementaire : De nombreuses normes (ISO 27001, RGPD, PCI-DSS) imposent une surveillance continue et une gestion efficace des événements de sécurité.

4. Réduction du temps de réponse : Une supervision efficace aide à réagir plus vite aux incidents, minimisant ainsi l'impact sur les opérations.

5. Amélioration de la posture de sécurité : Elle permet d'identifier les faiblesses du système et de mettre en place des mesures correctives.

Sans une supervision efficace, un audit de sécurité risque de révéler :

- Une incapacité à détecter les attaques en temps réel.
- Une absence de logs exploitables pour comprendre les incidents.
- Une non-conformité avec les standards de cybersécurité.
- Un temps de réaction trop long face aux menaces.

Proposition de mise en place de Nagios Core
Pour améliorer la supervision et répondre aux exigences de sécurité, la mise en place de Nagios Core est une solution pertinente.

Pourquoi choisir Nagios Core ?

- Solution open-source éprouvée : Nagios est l'un des outils de supervision les plus utilisés et bénéficie d'une large communauté.
- Surveillance multi-composants : Il permet de superviser les serveurs, équipements réseau, bases de données, applications et services critiques.
- Alertes en temps réel : Nagios envoie des notifications (email, SMS, etc.) en cas de défaillance ou d'anomalie.
- Personnalisation avancée : Grâce aux plugins, il est possible d'adapter Nagios aux besoins spécifiques de l'organisation.
- Interface web intuitive : Permet un suivi centralisé des statuts des équipements et services surveillés.

ETAPE 1 : INSTALLATION DE LA MACHINE UBUNTU

ETAPE 1

Choisir le lecteur « ubuntu-24-04-2 » dans « installer disc image file » puis choisir « Next ».

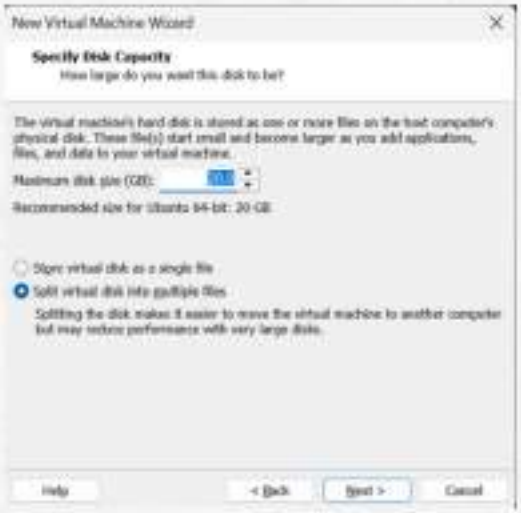
Cette étape permet d'aller plus vite et plus efficace à installer le disque Ubuntu.

ETAPE 2

Maintenant, vous changez le nom de la machine en mettant « Nagios-1 » et faire « Next ».

ETAPE 3

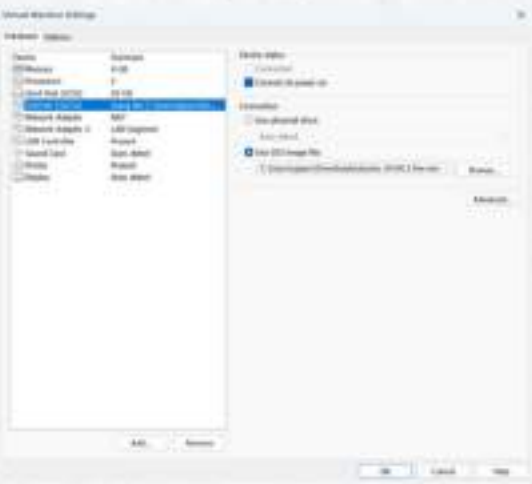
Pour le « Specify Disk Capacity », elle n'est pas à modifier, elle doit être à 20,0GB et ainsi elle doit rester dans l'option « Split virtual disk into multiple files », puis vous faites « Next ».

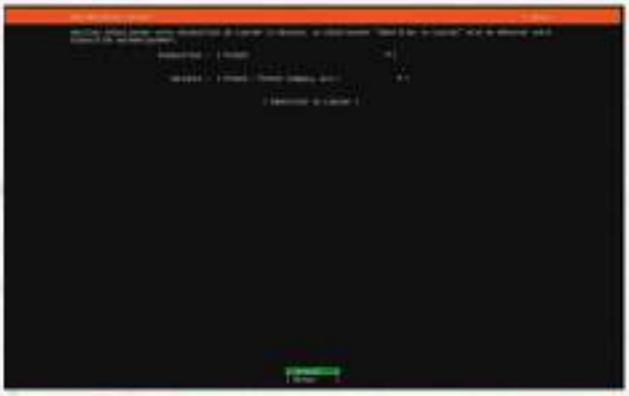


ETAPE 4

Après avoir créé la machine, allez sur les paramètres de la machine « Settings » :

Vous allez sur « CD/DVD(SATA) » pour ajouter le fichier ISO et ensuite vous cliquez sur « Use ISO image file » et choisir le fichier « ubuntu-24.04.2-live » puis vous faites « OK ».





ETAPE 5

Dans cette étape, vous allez allumer la machine Nagios, puis utiliser uniquement le clavier (sans souris) ainsi que vous allez sélectionner la langue en français et faire "Terminé".

ETAPE 6

Dans cette étape, vous allez sélectionner uniquement "Ubuntu Server" et faire "Terminé"

ETAPE 7

ATTENTION - Cette partie, vous allez juste sélectionner "Terminer" ! à ne pas changer !

Vous pouvez passer cette étape car on n'a pas de proxy à configurer

ETAPE 8

Cette partie permet de télécharger automatiquement de l'archive du miroir dont : "http://archive.ubuntu.com/ubuntu/" donc lorsqu'elle a finis de télécharger vous faites terminer.

ETAPE 9

Vous allez juste cocher "Utiliser un disque entier" et "Set up this disk as an LVM group" et faire terminer

ça c'est une sommaire du système de fichiers dont vous pouvez voir où est-ce qu'il y'a des erreurs ce que vous avez depuis le début avant de débuter le système, lorsque vous avez vérifier de A à Z, vous pouvez faire "Terminer"

ATTENTION !!! APRES VOUS NE POUVEZ PLUS RETOURNER APRES AVOIR CLIQUER SUR "TERMINER" !!!

ETAPE 10

Cette partie, vous allez compléter votre nom (Nagios), votre nom du serveur (nagios), votre nom d'utilisateur (user) et choisir un mot de passe (root).

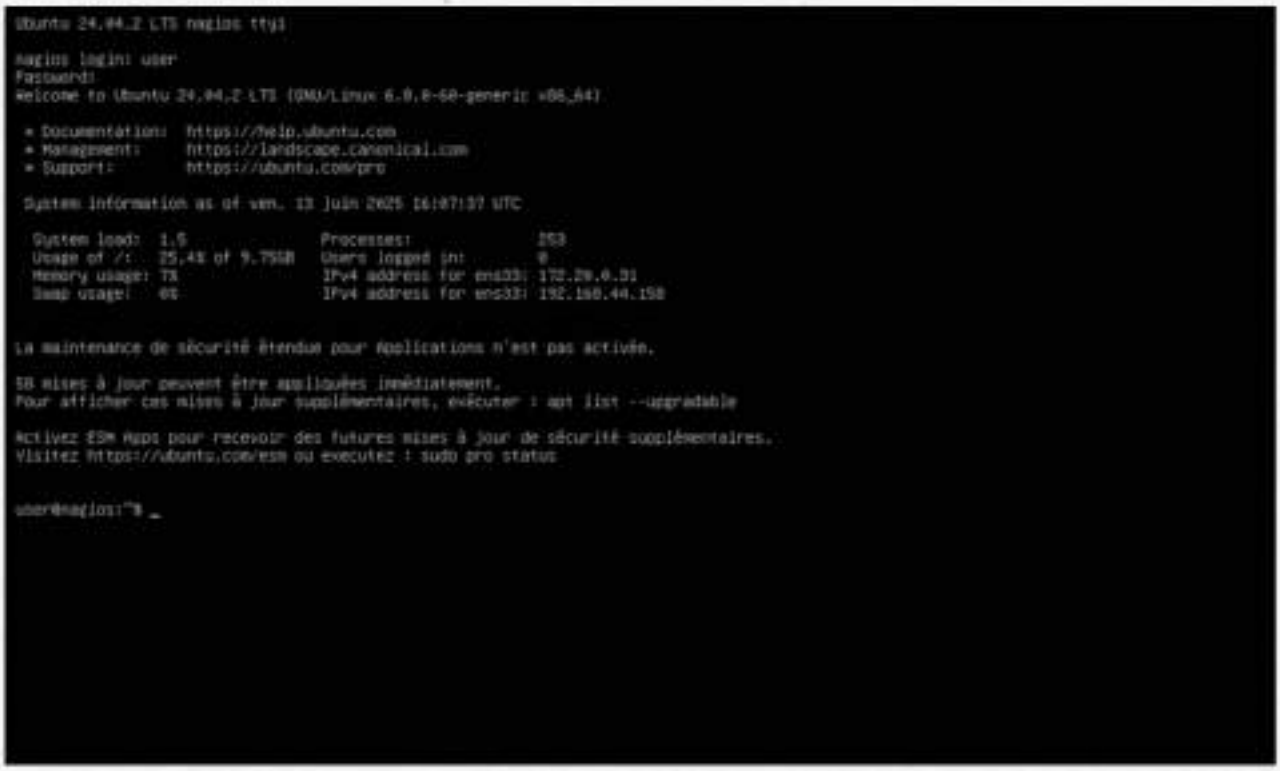
ETAPE 11

Vous pouvez a tout moment ignorer cette partie car nous avons pas Ubuntu PRO donc vous cochez "Skip for now" et faire Continuer

Pas de besoin d'installer le serveur OpenSSH donc vous faites "Terminer"

Cette partie, vous n'allez pas tout cocher donc vous allez faire "Terminer" !

ETAPE FINAL

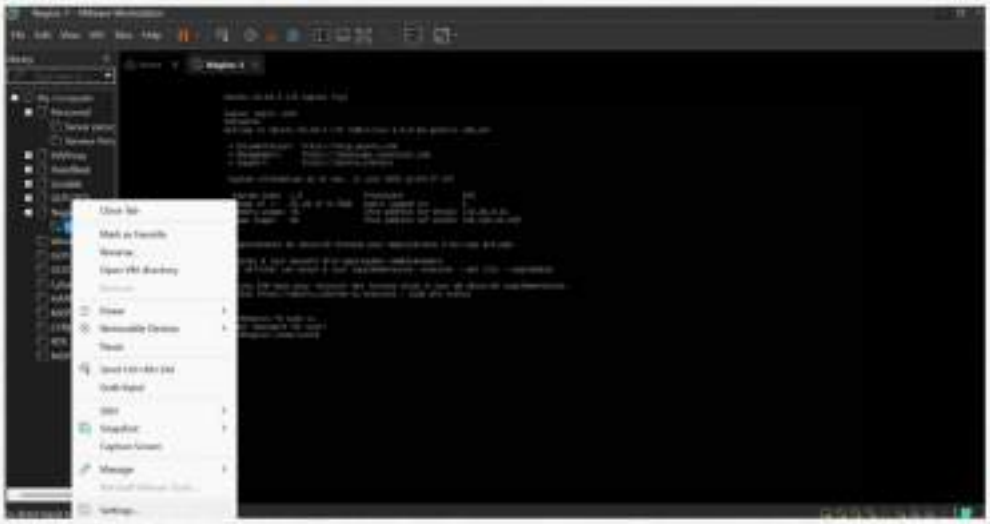


Maintenant, vous pouvez utiliser votre machine Ubuntu

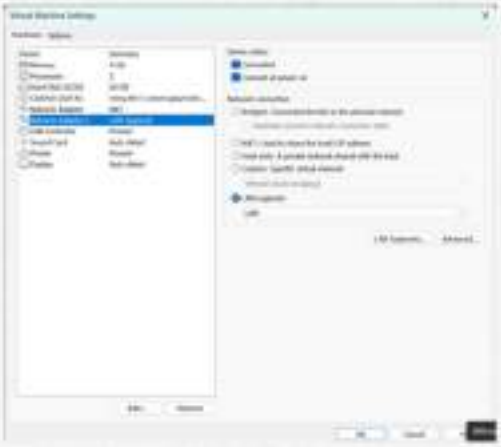
ETAPE 2 : CONFIGURATION DE LA MACHINE UBUNTU

ETAPE 1

Faite un clic droit sur votre machine "Nagios-1" et rendez-vous dans "Settings"



Maintenant, ajoutez une deuxième carte adaptateur et configurez-la en mode "LAN"



Après cela, vous pouvez redémarrer la machine en utilisant la commande "reboot"

```
user@nagios:~$ reboot_
```

ETAPE 2

Dans cette étape, vous devez mettre le compte en "root" en utilisant cette commande "sudo su".

```
user@nagios:~$ sudo su
[sudo] password for user:
root@nagios:/home/user#
```

ETAPE 3

Installer le paquet ifupdown (apt install ifupdown) pour configurer les interfaces réseaux

```
root@nagios:/home/user# apt install ifupdown_
```

ETAPE 4

Maintenant, vous allez sur le fichier interface en faisant :
nano /etc/network/interfaces

```
root@nagios:/home/user# nano /etc/network/interfaces
```

Après, vous allez remplir deux interfaces tel que :

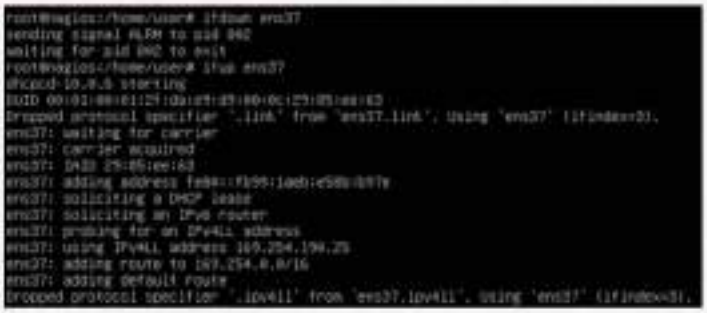
```
 GNU nano 7.2                                                    /etc/netwo
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source /etc/network/interfaces.d/*

auto ens33
iface ens33 inet static
address 172.20.0.31
netmask 255.255.255.0

auto ens37
iface ens37 inet dhcp
```

Ensuite, vous allez faire ifdown et ifup pour chaque ens

ens37



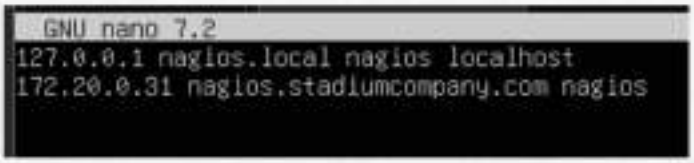
ens33

ETAPE 5

Vous allez mettre "nano /etc/hosts"

Cette commande sert à d'outil au niveau du système local pour mapper les noms d'hôtes aux adresses IP

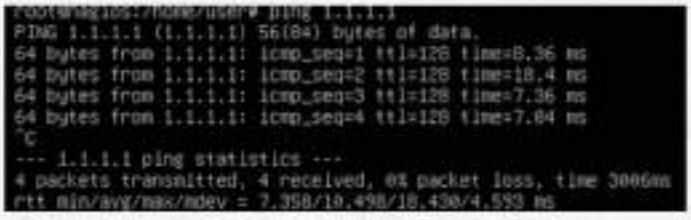
Maintenant, dans cette fichier, vous allez mettre ceux-ci :



ETAPE 6

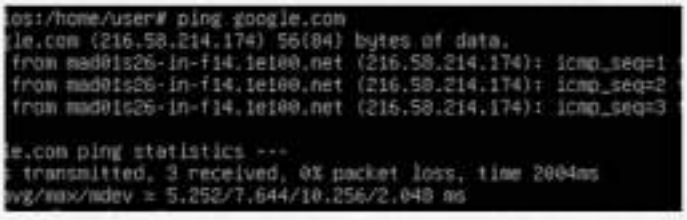
Maintenant, vous allez vérifier via un ping si l'adresse est bien en actif

ping 1.1.1.1



histoire de vérifier la passerelle par défaut

ping google.com



histoire de vérifier le DNS de la machine

ETAPE FINAL

Vous allez pouvoir installer les packages qui sont prérequis pour nos étapes suivante :

```
# apt update && apt upgrade -y
```

"apt update && apt upgrade -y"

Elle est utilisée sur les systèmes basés sur Ubuntu (comme Debian) pour gérer les mises à jour des logiciels via le terminal.

```
apt install -y autoconf gcc
```

"apt install -y autoconf gcc""

Elle sert à installer deux outils essentiels pour le développement logiciel sur les systèmes Linux basés sur Ubuntu.

```
# sudo apt install -y libc6 make wget unzip apache2 php libapache2-mod-php libgd-dev libssl-dev
```

"sudo apt install -y libc6 make wget unzip apache2 php libapache2-mod-php libgd-dev libssl-dev"

Elle installe un ensemble de paquets essentiels pour construire et faire tourner un serveur web sous Ubuntu. C'est souvent le point de départ pour des outils comme Nagios Core, Cacti ou d'autres solutions web auto-hébergées.

ETAPE 3 : INSTALLATION DE NAGIOS CORE

ETAPE 1

Maintenant, nous allons pouvoir télécharger nagios-core en faisant ceux-ci :

```
root@nagios:/home/user# cd /tmp
```

On va changer de directory en faisant "cd /tmd"

Ensuite, vous allez télécharger en faisant ceux-ci : wget -O nagioscore.tar.gz https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.5.8.tar.gz

```
root@nagios:/tmp# tar xvzf nagioscore.tar.gz_
```

Maintenant, décompressez le zip téléchargé : tar xvzf nagioscore.tar.gz

Ainsi que vous aller compiler les fichiers : cd nagioscore-nagios-4.5.8/

```
root@nagios:/tmp# cd nagioscore-nagios-4.5.8/
```

Puis vous allez taper : "./configure --with-httpd-conf=/etc/apache2/sites-enabled"

```
./configure --with-httpd-conf=/etc/apache2/sites-enabled
```

Cela lancera le script configure (dans le dossier nagioscore-nagios-4.5.0) avec l'option de configuration pour le répertoire de conf Apache.

Puis faites : "make all"

```
root@nagios:/tmp/nagioscore-nagios-4.5.8# make all
```

"make all" est une commande utilisée pour compiler un programme à partir de son code source.

ETAPE 2 :

Créez l'utilisateur et le groupe

Cela crée l'utilisateur et le groupe nagios. L'utilisateur www-data (correspondant à apache) est également ajouté au groupe nagios.

Vous allez taper cette commande : "make install-groups-users"
et aussi ça : "usermod -a -G nagios www-data"

```
root@nagios:/tmp/nagioscore-nagios-4.5.8# make install-groups-users
```

```
root@nagios:/tmp/nagioscore-nagios-4.5.8# usermod -a -G nagios www-data
```

ETAPE 3 :

Installez les binaires (exécutables)

Cette étape installe les fichiers binaires, les CGI et les fichiers HTML.

```
p/nagioscore-nagios-4.5.8# make install
```

Vous allez taper cette commande : "make install"

Installez le Service / Daemon

Cela installe les fichiers de service ou de démon et les configure également pour le démarrage automatique

```
root@nagios:/tmp/nagioscore-nagios-4.5.8# make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
```

"make install-init"

Les informations sur le démarrage et l'arrêt des services seront expliquées plus loin.

ETAPE 4 :

Installez le mode commande

```
root@nagios:/tmp/nagioscore-nagios-4.5.8# make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***
```

"make install-commandmode"

Cela installe et configure le fichier de commande externe.

ETAPE 5 :
Installez les Fichier de configuration

Dans cette étape, vous allez taper : "make install-config"

*Cela installe les fichiers de configuration * SAMPLE *. Celles-ci sont nécessaires car Nagios a besoin de quelques fichiers de configuration pour lui permettre de démarrer.*

ETAPE 6 :
Installez les fichier de configuration d'Apache

Cela installe les fichiers de configuration du serveur Web Apache et configure les paramètres Apache.

```
# make install-webconf
```

"make install-webconf"

```
# a2enmod rewrite
```

"a2enmod rewrite"

```
systemctl restart apache2
```

"systemctl restart apache2"

```
# a2enmod cgi
```

"a2emod cgi"

ETAPE 7 :
Configuration du Firewall (le pare-feu)

Vous allez autoriser le trafic entrant du port 80 sur le pare-feu local pour pouvoir accéder à l'interface Web de Nagios Core.

```
ufw allow Apache
```

"ufw allow Apache"

Cette commande permet à ton serveur d'accepter les connexions web nécessaires au bon fonctionnement de l'interface Nagios dans un navigateur.

```
# ufw reload
```

"ufw reload"

Il permet de recharger les règles du pare-feu UFW
il utilise si on a modifié manuellement des fichiers de configuration du pare-feu
Il reprend toutes les règles sans redémarrer le pare-feu

ETAPE 8 :
La creation "nagiosadmin User Account"

On va créer un compte utilisateur Apache pour pouvoir vous connecter à Nagios.

La commande suivante créera un compte utilisateur appelé nagiosadmin et vous serez invité à fournir un mot de passe pour le compte.

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

On est train de créer un utilisateur nommé nagiosadmin avec un mot de passe, pour qu'il puisse se connecter à l'interface web de Nagios Core, protégée par Apache.
Après cette commande, Apache demandera un nom d'utilisateur et un mot de passe pour accéder à la page de Nagios.

ETAPE 9 :
Redémarrage Apache2

```
systemctl restart apache2.service
```

```
systemctl status apache2.service
```

Voici le résultat, il est bien en "active running"

```
root@nagios:/tmp# systemctl status apache2.service
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-07-02 20:47:18 UTC; 3min 17s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 1542 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 1570 (apache2)
      Tasks: 6 (limit: 4540)
     Memory: 17.9M (peak: 18.3M)
        CPU: 239ms
     CGroup: /system.slice/apache2.service
             ├─1570 /usr/sbin/apache2 -k start
             ├─1572 /usr/sbin/apache2 -k start
             ├─1573 /usr/sbin/apache2 -k start
             ├─1574 /usr/sbin/apache2 -k start
             ├─1575 /usr/sbin/apache2 -k start
             └─1576 /usr/sbin/apache2 -k start

juil. 02 20:47:18 nagios systemd[1]: Starting apache2.service - The Apache HTTP Server...
juil. 02 20:47:18 nagios systemd[1]: Started apache2.service - The Apache HTTP Server.
```

ETAPE 10 :
Démarrage Nagios

```
systemctl restart nagios.service
```

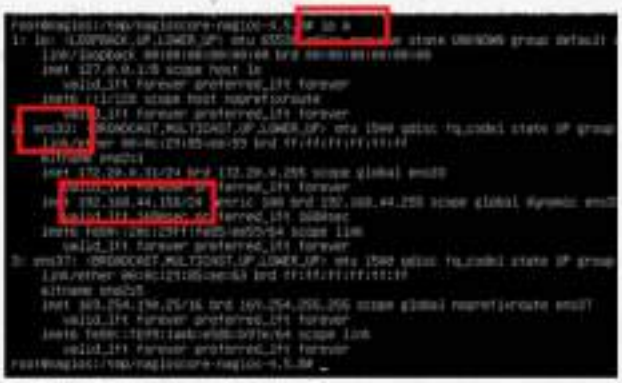
```
systemctl status nagios.service
```

Voici le résultat, il est bien en "active running"

ETAPE 11 :
Test

Nagios est maintenant en cours d'exécution, pour confirmer cela, vous devez vous connecter à l'interface Web de Nagios.

Ouvrez n'importe quel navigateur web, comme ici par exemple : Google Chrome



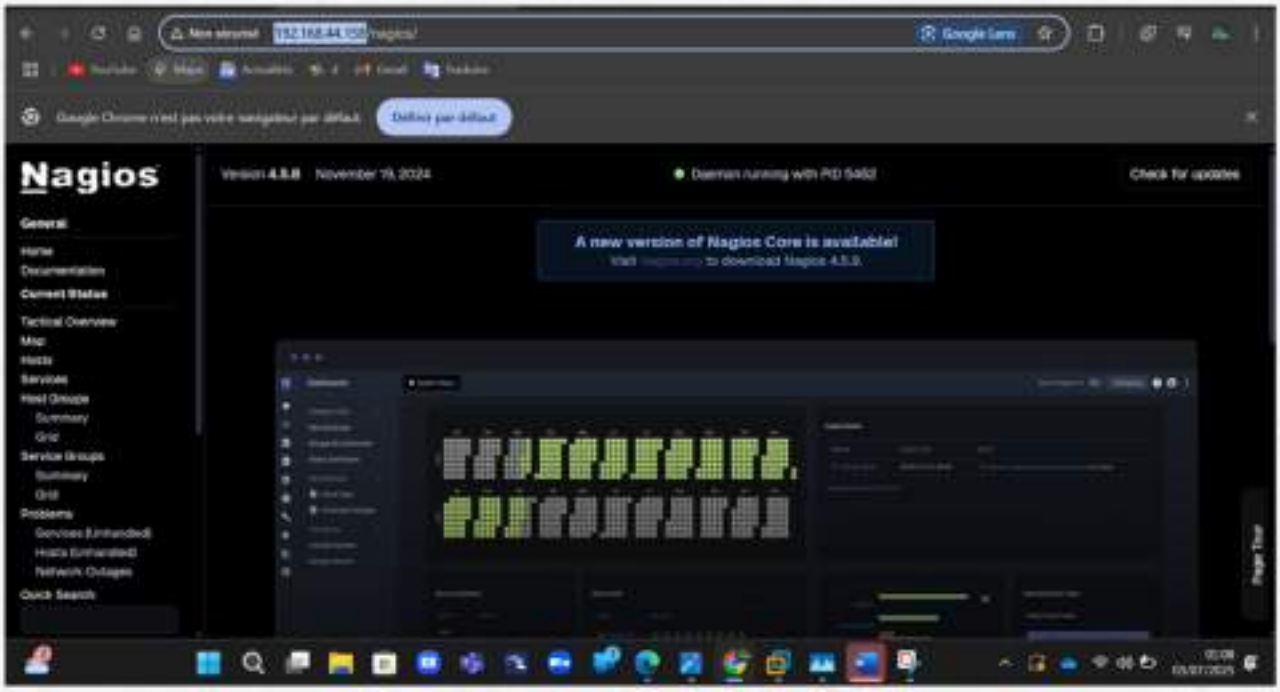
Accédez à votre machine Nagios-1, tapez la commande ip a, puis repérez l'adresse IP de l'interface réseau ens33. Dans cet exemple, l'adresse IP est 192.168.44.158.

Une fois l'adresse IP de l'interface réseau ens33 identifiée, ouvrez votre navigateur web et saisissez cette adresse IP dans la barre d'adresse. Vous devriez alors accéder à l'interface web de Nagios, comme illustré ci-dessous.

Comme le service fonctionne correctement, ajoutez simplement /nagios à la fin de l'adresse IP dans la barre d'adresse de votre navigateur. Par exemple : http://192.168.44.158/nagios.

Une fois l'adresse saisie avec /nagios, vous serez redirigé vers un formulaire de connexion à l'interface web de Nagios. Entrez les identifiants suivants : Nom d'utilisateur : nagiosadmin Mot de passe : root (il s'agit du mot de passe que vous avez défini lors de la création du compte nagiosadmin).

Félicitations, vous avez installé Nagios Core !



ETAPE 4 : INSTALLATION DES PLUGINS NAGIOS

Nagios Core a besoin de plugins pour fonctionner correctement. Les étapes suivantes vous guideront tout au long de l'installation des plugins Nagios.

Ces étapes isr que les étapes suivantes installent la plupart des plugins fournis dans le package Nagios Plugins.

Cependant, certains plugins nécessitent d'autres bibliothèques qui ne sont pas incluses dans ces instructions.

Veuillez consulter les articles de la base de connaissances pour obtenir deapts instructions d'installation détaillées:

ETAPE 1 :
Prérequis

Installer les packages prérequis :

```
isor (qemu) binaries on this host.
# apt install -y libmcrypt-dev libssl-dev bc gawk dc build-essential snmp libnet-snmp-perl gettext
```

Voici la commande pour procéder à cette étape:
"apt install -y libmcrypt-dev libssl-dev bc gawk dc build-essential snmp libnet-snmp-perl gettext

ETAPE 2 :
Téléchargement de la source :

```
cd /tmp_
```

1) Dans cette étape, vous allez taper ceux-ci :
"cd /tmp"

```
wget --no-check-certificate -O nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz
```

2) Puis, vous allez taper ceux-ci :
"wget --no-check-certificate -O nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz"

ETAPE 3 :
Décompresser la source

```
tar zxvf nagios-plugins.tar.gz_
```

Cette commande font partie du processus d'installation des plugins Nagios, un système de supervision de services réseau.

On entre dans le dossier de travail, puis on décompresse les fichiers nécessaires à l'installation.

ETAPE 4 :
Compilation + Installation

```
cd nagios-plugins-release-2.2.1/
```

Vous entrez dans le dossier extrait contenant le code source des plugins.

```
# ./tools/setup_
```

Elle prépare les scripts de configuration (optionnel mais parfois utile).

```
./configure
```

Elle vérifie votre système pour s'assurer qu'il peut compiler les plugins (vérifie les dépendances, crée les Makefiles, etc.).

```
# make install_
```

Elle compile et installe les plugins sur le système.

ETAPE 5:
TEST

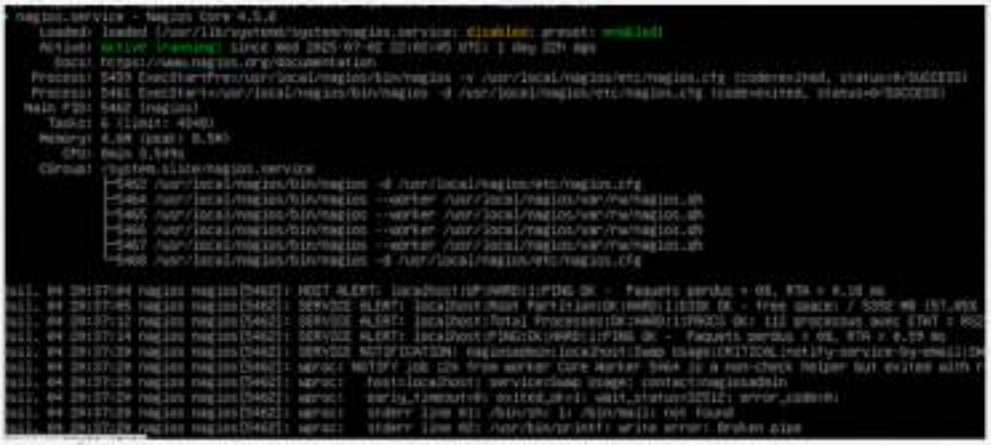
```
systemctl start nagios.service
```

Elle permet démarrer le service Nagios.

Elle permet vérifie que tout fonctionne bien (vous devais voir "active (running)").

```
systemctl status nagios.service
```

Test réussi



ETAPE 5 : SUPERVISION DES HÔTES

1) Activation de la supervision des machines et équipements réseau

Editer le fichier nagios.cfg

```
# nano -c /usr/local/nagios/etc/nagios.cfg
```

Le fichier nagios.cfg est le cœur de la configuration de Nagios Core. Il détermine :

- Les fichiers de ressources utilisés
- Les répertoires contenant les définitions d'hôtes et de services
- Les options de logging, notifications, planification, etc.

```
# Definitions for monitoring a Windows machine
cfg_file=/usr/local/nagios/etc/objects/windows.cfg
```

Afin de superviser les machines Windows, décommenter (enlever le #) la ligne 38

```
# Definitions for monitoring a router/switch
cfg_file=/usr/local/nagios/etc/objects/switch.cfg
```

Afin de superviser les machines switchs, routeurs, et équipements d'interconnexions, décommenter (enlever le #) la ligne 41

```
# Definitions for monitoring a network printer
cfg_file=/usr/local/nagios/etc/objects/printer.cfg
```

Afin de superviser les machines imprimantes, décommenter (enlever le #) la ligne 44

Enregistrer le fichier en faisant CRTL + X

2) Ajout et supervision des serveurs Windows

```
nano -c /usr/local/nagios/etc/objects/windows.cfg
```

Dans la section « HOST DEFINITIONS », aller à la ligne 24, section <define host> et écrire en face de 'host_name' le nom de la machine 'DC', dans alias le pseudonyme que l'on veut donner à la machine, et dans address l'adresse ip de la machine tel que :

```
# Change the host_name, alias, and address to fit your situation

define host {

    use                     windows-server          ; Inherit default values from a template
    host_name               DC              ; The name we're giving to this host
    alias                   AD-DS       ; A longer name associated with the host
    address                 172.20.0.10_            ; IP address of the host
}
```

Important : Ne pas oublier de changer le host_name des différents services (dans les blocs 'define service' un peu plus bas dans le même fichier) en supprimant"winserver"et en ajoutant les noms de chaque serveurs (dans notre cas DC).

65 : Bloc de supervision de l'agent de supervision NSClient

```
    # Create a service for monitoring the version of NSClient++ that is installed
    # Change the host_name to match the name of the host you defined above

    define service {

        use                     generic-service
        host_name               DC
        service_description     NSClient++ Version
        check_command           check_nt!CLIENTVERSION
    }
```

79 : Bloc de supervision du temps depuis lequel la machine tourne sans interruption.

```
# Create a service for monitoring the uptime of the server
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC_
    service_description     Uptime
    check_command           check_nt!UPTIME
}
```

92 : Bloc de supervision du charge système, une mesure de la quantité de travail

```
# Create a service for monitoring CPU load
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC
    service_description     CPU Load
    check_command           check_nt!CPULOAD!-l 5,80,90
}
```

105 : Bloc de supervision de l'utilisation de la mémoire.

```
# Create a service for monitoring memory usage
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC
    service_description     Memory Usage
    check_command           check_nt!MEMUSE!-w 80 -c 90
}
```

118 : Bloc de supervision de l'espace disque

```
# Create a service for monitoring C:\ disk usage
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC
    service_description     C:\ Drive Space
    check_command           check_nt!USEDDISKSPACE!-l c -w 80 -c 90
}
```

131 : Bloc de supervision du service de publication World Wide Web : w3svc

```
# Create a service for monitoring the W3SVC service
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC
    service_description     W3SVC
    check_command           check_nt!SERVICESTATE!-d SHOWALL -l W3SVC
}
```

143: Bloc de supervision d'Explorer

```
# Create a service for monitoring the Explorer.exe process
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC
    service_description     Explorer
    check_command           check_nt!PROCSTATE!-d SHOWALL -l Explorer.exe
}
```

On peut ajouter plusieurs serveurs sur le même service, il suffit de séparer leurs noms par une virgule, on peut aussi surveiller d'autre service en ajoutant des blocs service.

Maintenant, nous allons redémarrer le service nagios en faisant avec cette commande
service nagios restart

```
root@nagios:/tmp/nagios-plugins-release-2.2.1# service nagios restart
```

3) Ajout et supervision des serveurs Linux

Maintenant on va editer le fichier localhost.cfg pour ajouter des serveurs Linux, en faisant cette commande :

```
nano -c /usr/local/nagios/etc/objects/localhost.cfg_
```

Dans la section « HOST DEFINITIONS », aller à la ligne 24, section <define host> et écrire dans host_name le nom de la machine Linux, dans alias le pseudonyme que l'on veut donner à la machine, et dans « address » l'adresse ip de la machine tel que :

Juste après ce premier bloc, ajoutez le bloc suivant :

```
define host {

        use             linux-server
        host_name       Zimbra
        alias           Serveur de messagerie
        address         172.20.0.30
}
```

Changez localhost dans members du bloc define hostgroup par Nagios,Zimbra

Important : Ne pas oublier de changer le host_name en suppriment "localhost" et en ajoutant les noms de chaque serveurs pour chaque « define service ». On peut ajouter plusieurs serveurs sur le même service, il suffit de séparer leurs noms par une virgule.

```
define service {
    use                     local-service         ; Name of service template to use
    host_name               Nagios, Zimbra
    service_description     PING
    check_command           check_ping!100.0,20%!500.0,60%
}
```

```
# Define a service to check the disk space of the root partition
# on the local machine.  Warning if < 20% free, critical if
# < 10% free space on partition.

define service {
    use                     local-service         ; Name of servic
    host_name               Nagios, Zimbra_
    service_description     Root Partition
    check_command           check_local_disk!20%!10%!/
}
```

```
# Define a service to check the number of currently logged in
# users on the local machine.  Warning if > 20 users, critical
# if > 50 users.

define service {
    use                     local-service         ; Name of servi
    host_name               Nagios, Zimbra
    service_description     Current Users
    check_command           check_local_users!20!50
}
```

```
# Define a service to check the number of currently running procs
# on the local machine.  Warning if > 250 processes, critical if
# > 400 processes.

define service {
    use                     local-service         ; Name of service te
    host_name               Nagios, Zimbra
    service_description     Total Processes
    check_command           check_local_procs!250!400!RSZDT
}
```

```
# Define a service to check the load on the local machine.

define service {
    use                     local-service         ; Name
    host_name               Nagios, Zimbra_
    service_description     Current Load
    check_command           check_local_load!5.0,4,0,3.0!:
}
```

```
# Define a service to check the swap usage the local machine.
# Critical if less than 10% of swap is free, warning if less than

define service {
    use                     local-service         ; Name of ser
    host_name               Nagios, Zimbra
    service_description     Swap Usage
    check_command           check_local_swap!20%!10%
}
```

```
# Define a service to check SSH on the local machine.
# Disable notifications for this service by default, as not all

define service {
    use                     local-service         ; Name of se
    host_name               Nagios, Zimbra_
    service_description     SSH
    check_command           check_ssh
    notifications_enabled   0
}
```

```
# Define a service to check HTTP on the local machine.
# Disable notifications for this service by default, as not all use

define service {
    use                     local-service         ; Name of servi
    host_name               Nagios, Zimbra
    service_description     HTTP
    check_command           check_http
    notifications_enabled   0
}
```

ETAPE 6 : CONFIGURATION DES NOTIFICATIONS PAR MAIL

Maintenant, vous allez installer les paquets sendmail, mailutils et ssmtp :

```
apt install sendmail
```

```
apt install mailutils
```

```
apt install ssmtp
```

Ainsi que, vous allez configurer le fichier du SSMTP:

```
nano -c /etc/ssmtp/ssmtp.conf
```

Modifier la ligne 10:
mailhub=mail.stadiumcompany.com

```
# The place where the mail goes. The actual machine name is required no
# MX records are consulted. Commonly mailhosts are named mail.domain.com
mailhub=mail.stadiumcompany.com
```

Puis enregistrer, cette fichier en faisant "CRTL + X"

Dans un second temps, nous allons configurer
de l'adresse de messagerie du compte root :

```
nano -c /etc/ssmtp/revaliases
```

Puis, vous allez rajouter cette ligne :
root:adminNagios@stadiumcompany.com

```
# sSMTP aliases
#
# Format:       local_account:outgoing_address:mailhub
#
# Example: root:your_login@your.domain:mailhub.your.domain[:port]
# where [:port] is an optional port number that defaults to 25.
root:adminNagios@stadiumcompany.com
```

Attention :
La machine nagios doit pouvoir résoudre le nom : mail.stadiumcompany.com :

1- Allumez votre contrôleur de domaine stadiumcompany.com : 172.20.0.10 ou autre
2- Renseigner à nagios son serveur DNS 172.20.0.10

```
# nano /etc/resolv.conf
```

```
nameserver 172.20.0.10
nameserver 1.1.1.1
search stadiumcompany.com
```

Testez la résolution de nom depuis nagios :

```
nslookup mail.stadiumcompany.com
```

Voici le résultat :

Maintenant, vous allez redémarrer le service nagios : service nagios restart
· Testez l'envoi du mail : echo "Contenu du mail" | mail -s "Sujet 1" admin@stadiumcompany.com
Ou à votre adresse mail perso : echo "Contenu du mail" | mail -s "Sujet 1" MailPerso

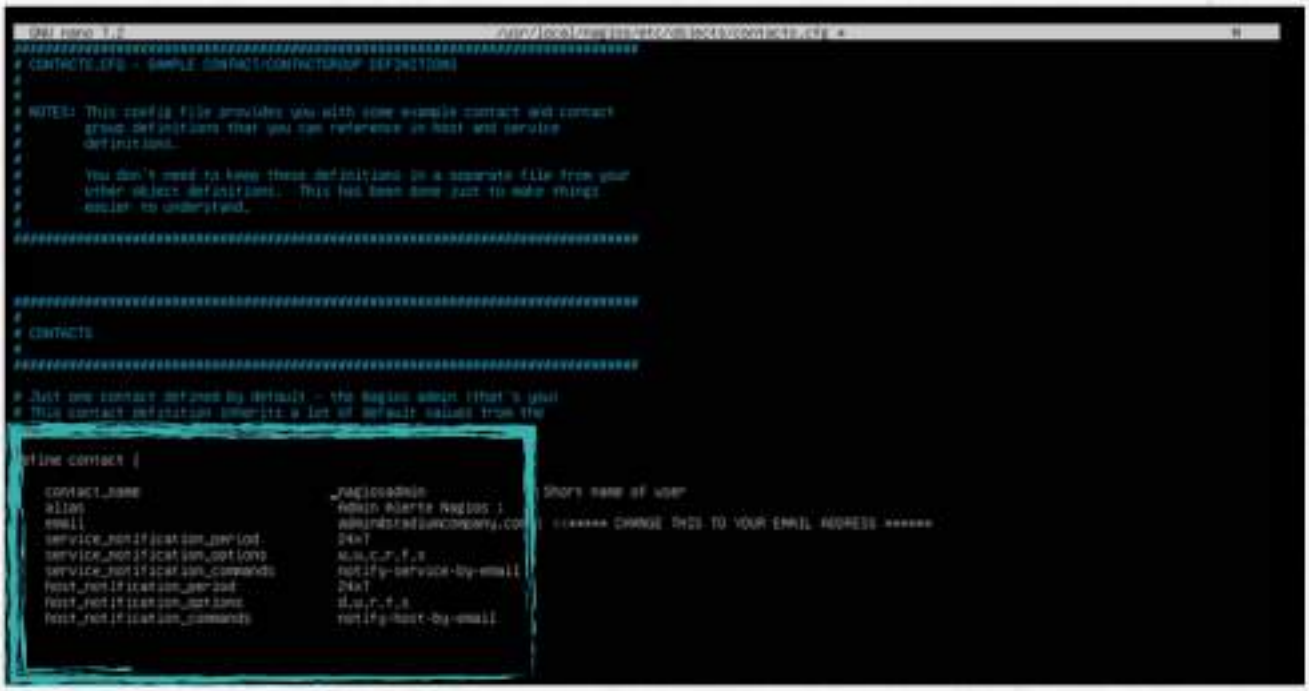
Vérifiez l'obtention du mail dans la boite de messagerie admin@stadiumcompany.com

Maintenant nous allons modifier du fichier contact.cfg grâce à la commande :

nano -c /usr/local/nagios/etc/objects/contact.cfg

Plus, vous allez mettre ceux-ci dans le fichier :

```
define contact{
                contact_name                    nagiosadmin
                alias                   Admin Alerte Nagios
                email admin@stadiumcompany.com
                service_notification_period         24x7
                service_notification_options  w,u,c,r,f,s
        service_notification_commands   notify-service-by-email
                host_notification_period 24x7
                host_notification_options d,u,r,f,s
        host_notification_commands notify-host-by-email
                }
```



EXPLICATION

contact_name = nom du contact.

alias = description du contact.

email = adresse mail du contact.

service_notification_period 24x7 = période d'envoi des notification pour les services (applications) 7j/j 24h/24.

service_notification_options w,u,c,r,f,s = notifie les options choisis pour les services (w : informe les états de service WARNING, u : informe sur les états de service UNKNOWN, c : informe les états de service CRITICAL, r : informe le service RECOVERY (états OK), f : informe lorsque le service démarre et arrête FLAPPING, n : ne pas notifier le contact sur tout type de notifications de service).

Service_notification_commands notify-service-by-email = choix d'être notifier par email sur l'état des services.

Host_notification_period 24x7 =période d'envoi des notifications pour les hôtes (pc) 7j/j 24h/24

Host_notification_options d,u,r,f,s = notifie les options choisis pour les hôtes (d: informe sur le statut DOWN de l'hôte, u : informe sur le statut UNREACHABLE de l'hôte,r : informe sur l'hôte RECOVERY (états allumé), f : informe au démarrage de l' hôte et arrête FLAPPING, s : Envoie des notifications lorsque l'hôte ou le service prévu les temps d'arrêt commence et se termine, n(none) : Ne pas notifier le contact sur tout type de notifications d'hôtes.

Host_notification_commands notify-host-by-email = choix d'être notifier par email sur l'état des services.

ETAPE FINAL – VERIFICATION DE LA RECEPTION DES NOTIFICATIONS

Dans cette étape, vous allez redémarrer le service Nagios grâce à cette commande :

```
service nagios restart
```

Autoriser l'envoi de notification :

Aller dans le dossier /usr/local/nagios/etc/objects/ :

cd /usr/local/nagios/etc/objects/

```
cd /usr/local/nagios/etc/objects/
```

et rajouter les lignes suivant dans chaque sections « define host{ } » et dans chaque sections « define service{ } » des fichiers /usr/local/nagios/etc/objects/windows.cfg, /usr/local/nagios/etc/objects/localhost.cfg et /usr/local/nagios/etc/objects/switch.cfg:

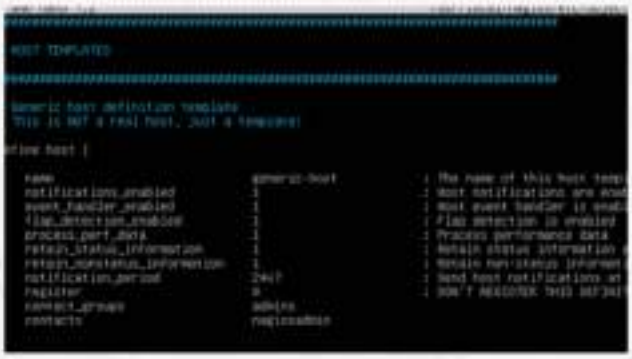
contact_groups admins
contacts nagiosadmin

Dans le fichier, /usr/local/nagios/etc/objects/windows.cfg :

Il faut ajouter :
contact_groups admins
contacts nagiosadmin

Dans chaque "define host" et "define service" !

Dans le fichier, /usr/local/nagios/etc/objects/localhost.cfg :



```
# Local service definition template
# This is NOT a real service, just a template!

define service {

    name                            local-service
    use                             generic-service
    max_check_attempts              4
    check_interval                  5
    retry_interval                  1
    register                        0
    contact_groups                  admins
    contacts                        nagiosadmin
}
```

Il faut ajouter :
contact_groups admins
contacts nagiosadmin

Dans chaque bloc de "define host" et "define service" !

Dans le fichier, /usr/local/nagios/etc/objects/switch.cfg :

```
# Define the switch that we'll be monitoring

define host {

    use                             generic-switch
    host_name                       linksys-srw224p
    alias                           Linksys SRW224P Switch
    address                         192.168.1.253
    hostgroups                      switches
    contact_groups                  admins
    contacts                        nagiosadmin
}
```

```
# Monitor uptime via SNMP

define service {

    use                             generic-service
    host_name                       linksys-srw224p
    service_description             Uptime
    check_command                   check_snmp!-C public -o sysUpTime.0
    contact_groups                  admins
    contacts                        nagiosadmin
}
```

Il faut ajouter :
contact_groups admins
contacts nagiosadmin

Dans chaque bloc de "define host" et "define service" !

Maintenant, vous allez dans le fichier commands.cfg :

nano –c /usr/local/nagios/etc/objects/commands.cfg

et aller à la ligne 29 et 37 et modifier le fichier pour que /usr/bin/mail soit présent (compléter uniquement par /usr):

```
# notify-host-by-email' command definition
define command {
command_name notify-host-by-email
command_line /usr/bin/printf "%b" "***** Icinga *****\n\n \Notification Type:$NOTIFICATIONTYPE$\n \Host:
$HOSTNAME$\n \State: $HOSTSTATE$\n \Address: $HOSTADDRESS$\n \Info: $HOSTOUTPUT$\n\n \Date/Time:
$LONGDATETIME$\n" \ | /usr/bin/mail-s \"** $NOTIFICATIONTYPE$ Host Alert: \$HOSTNAME$ is $HOSTSTATE$ **"
\$CONTACTEMAIL$
}

# 'notify-service-by-email' command definition
define command {
command_name notify-service-by-email
command_line /usr/bin/printf "%b" "***** Icinga *****\n\n \Notification Type: $NOTIFICATIONTYPE$\n\n \Service:
$SERVICEDESC$\n \Host: $HOSTALIAS$\n \Address: $HOSTADDRESS$\n \State: $SERVICESTATE$\n\n \Date/Time:
$LONGDATETIME$\n\n \Additional Info:\n\n \$SERVICEOUTPUT$\n" \ | /usr/bin/mail-s \"** $NOTIFICATIONTYPE$ Service Alert
\$HOSTALIAS$/$SERVICEDESC$ is $SERVICESTATE$ **" \$CONTACTMAIL$
}
```

Redémarrer le service nagios :

service nagios restart

```
service nagios restart
```

Vérifiez la réception de notification

Mail reçu, c'est gagné ;-)