420-S0R-TT

Surveillance et optimisation des réseaux

Notes de cours

Installation de Nagios 4 sur Alma Linux 9.4

Été-2024

Table des matières

1 Introduction

Dans ce module on va installer et configurer Nagios 4 sur Alma Linux 9.4.

Nagios est configuré à l'aide de fichiers texte et fournit une interface Web intuitive pour l'administration et la surveillance. En cas de problème, de dégradation du service ou de panne, Nagios enverra des alertes par courriel, SMS ou même un appel téléphonique s'il est configuré pour le faire.

2 Désactiver SeLinux

Configurer SeLinux en mode permissive :

```
[root@localhost ~]# sed -i 's/SELINUX=.*/SELINUX=permissive/g'
/etc/selinux/config
```

```
[root@localhost ~]# setenforce 0
setenforce: SELinux is disabled
```

3 Prérequis

Faire la mise à jour de votre serveur Linux.

```
[root@localhost ~]# dnf -y update
```

Installer les paquetages qui sont prérequis.

```
[root@localhost ~]# dnf -y install @php @perl @httpd
```

```
[root@localhost ~]# dnf -y install unzip glibc automake glibc-common gettext autoconf php php-cli gcc gd gd-devel net-snmp openssl-devel unzip net-snmp postfix net-snmp-utils
```

```
[root@localhost ~]# dnf -y groupinstall "Development Tools"
```

Démarrer et activer les services httpd et php-fpm.

```
[root@localhost ~]# systemctl enable --now httpd php-fpm

Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service →
/usr/lib/systemd/system/httpd.service.
Created symlink /etc/systemd/system/multi-user.target.wants/php-fpm.service →
/usr/lib/systemd/system/php-fpm.service.
```

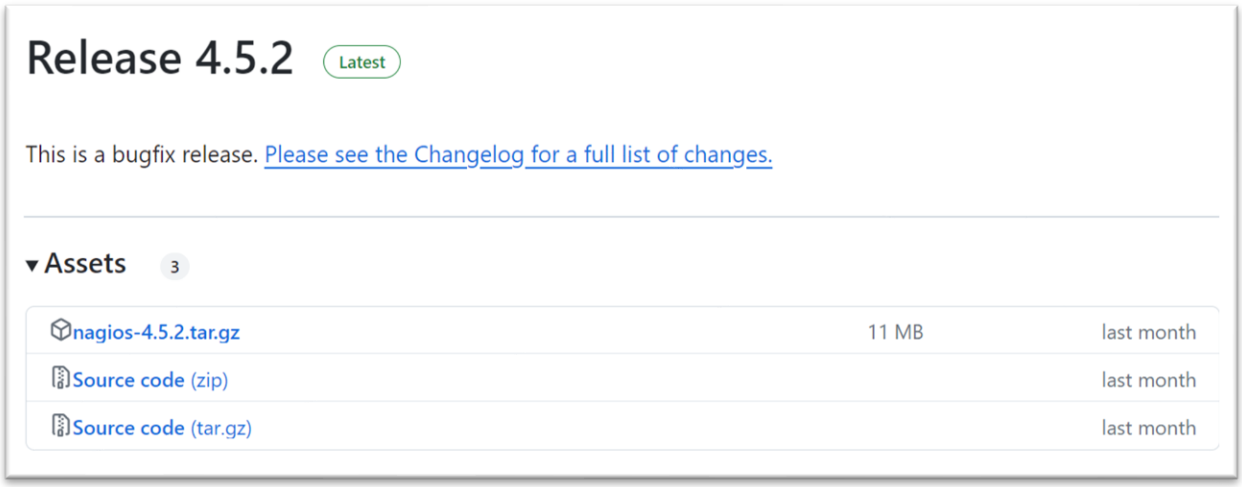
Vérifier le statut des services httpd et php-fpm.

```
[root@localhost ~]# systemctl status httpd php-fpm

● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: di>
    Drop-In: /usr/lib/systemd/system/httpd.service.d
             └─php-fpm.conf
     Active: active (running) since Tue 2024-05-28 19:56:05 EDT; 35s ago
       Docs: man:httpd.service(8)
   Main PID: 49962 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>
      Tasks: 177 (limit: 10899)
     Memory: 28.3M
        CPU: 133ms
     CGroup: /system.slice/httpd.service
             ├─49962 /usr/sbin/httpd -DFOREGROUND
             ├─49969 /usr/sbin/httpd -DFOREGROUND
             ├─49970 /usr/sbin/httpd -DFOREGROUND
             ├─49971 /usr/sbin/httpd -DFOREGROUND
             └─49972 /usr/sbin/httpd -DFOREGROUND

May 28 19:56:05 localhost systemd[1]: Starting The Apache HTTP Server...
May 28 19:56:05 localhost httpd[49962]: AH00558: httpd: Could not reliably dete>
May 28 19:56:05 localhost httpd[49962]: Server configured, listening on: port 80
May 28 19:56:05 localhost systemd[1]: Started The Apache HTTP Server.
```

4 Télécharger Nagios Core

Consulter la page des versions pour identifier la dernière version disponible de Nagios.

https://github.com/NagiosEnterprises/nagioscore/releases




```
[root@localhost ~]# export VER="4.5.2"
```

Télécharger et extraire le fichier tar Nagios.

```
[root@localhost ~]# curl -SL
https://github.com/NagiosEnterprises/nagioscore/releases/download
/nagios-$VER/nagios-$VER.tar.gz | tar -xzf -

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 11.0M  100 11.0M    0     0  11.8M      0 --:--:-- --:--:-- --:--:-- 11.8M
```

Se déplacer sur le répertoire source de Nagios.

```
[root@localhost ~]# cd nagios-$VER
[root@localhost nagios-4.5.2]#
```

5 Compiler Nagios Core

1) Exécuter le script configure.

```
[root@localhost nagios-4.5.2]# ./configure

checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
...
checking for unzip... /usr/bin/unzip
configure: creating ./config.status
config.status: creating html/index.php
config.status: creating Makefile
config.status: creating lib/Makefile
...
config.status: creating lib/iobroker.h

Creating sample config files in sample-config/ ...


*** Configuration summary for nagios 4.5.2 2024-04-30 ***:

 General Options:
 -------------------------
        Nagios executable:  nagios
        Nagios user/group:  nagios,nagios
       Command user/group:  nagios,nagios
             Event Broker:  yes
        Install ${prefix}:  /usr/local/nagios
    Install ${includedir}:  /usr/local/nagios/include/nagios
                Lock file:  /run/nagios.lock
   Check result directory:  /usr/local/nagios/var/spool/checkresults
          Init directory:  /lib/systemd/system
  Apache conf.d directory:  /etc/httpd/conf.d
             Mail program:  /usr/sbin/sendmail
                  Host OS:  linux-gnu
          IOBroker Method:  epoll

 Web Interface Options:
 ------------------------
                 HTML URL:  http://localhost/nagios/
                  CGI URL:  http://localhost/nagios/cgi-bin/
 Traceroute (used by WAP):


Review the options above for accuracy.  If they look okay,
type 'make all' to compile the main program and CGIs.
```

2) Exécuter la commande make avec l'option all pour compiler les programmes sources.

```
[root@localhost nagios-4.5.2]# make all

...
*** Compile finished ***

If the main program and CGIs compiled without any errors, you
can continue with testing or installing Nagios as follows (type
'make' without any arguments for a list of all possible options):

  make test
     - This runs the test suite

  make install
     - This installs the main program, CGIs, and HTML files

  make install-init
     - This installs the init script in /lib/systemd/system

  make install-daemoninit
     - This will initialize the init script
       in /lib/systemd/system

  make install-groups-users
     - This adds the users and groups if they do not exist

  make install-commandmode
     - This installs and configures permissions on the
       directory for holding the external command file

  make install-config
     - This installs *SAMPLE* config files in /usr/local/nagios/etc
       You'll have to modify these sample files before you can
       use Nagios.  Read the HTML documentation for more info
       on doing this.  Pay particular attention to the docs on
       object configuration files, as they determine what/how
       things get monitored!

  make install-webconf
     - This installs the Apache config file for the Nagios web interface

  make install-exfoliation
     - This installs the Exfoliation theme for the Nagios web interface

  make install-classicui
     - This installs the classic theme for the Nagios web interface


*** Support Notes *******************************************

If you have questions about configuring or running Nagios, please make sure
that you:

     - Look at the sample config files
```

```
     - Read the documentation on the Nagios Library at:
          https://library.nagios.com

before you post a question to one of the mailing lists.
Also make sure to include pertinent information that could
help others help you.  This might include:

     - What version of Nagios you are using
     - What version of the plugins you are using
     - Relevant snippets from your config files
     - Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:

       https://support.nagios.com

************************************************************

Enjoy.

[root@localhost nagios-4.5.2]#
```

3) Créer le groupe et l'utilisateur nagios

Exécuter la commande make avec l'option install-groups-users pour créer le groupe nagios et l'utilisateur nagios pour la console web de Nagios.

```
[root@localhost nagios-4.5.2]# make install-groups-users

groupadd -r nagios
useradd -g nagios nagios
```

Ajouter l'utilisateur apache au groupe nagios.

```
[root@localhost nagios-4.5.2]# usermod -a -G nagios apache
```

4) Installer Nagios Core

Exécuter la commande make avec l'option install pour installer Nagios Core.

```
[root@localhost nagios-4.5.2]# make install

...
*** Main program, CGIs and HTML files installed ***

You can continue with installing Nagios as follows (type 'make'
without any arguments for a list of all possible options):

  make install-init
     - This installs the init script in /lib/systemd/system

  make install-commandmode
     - This installs and configures permissions on the
       directory for holding the external command file

  make install-config
     - This installs sample config files in /usr/local/nagios/etc

make[1]: Leaving directory '/root/nagios-4.5.2'
```

5) Installer le script init dans /lib/systemd/system.

```
[root@localhost nagios-4.5.2]# make install-daemoninit

/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
Created symlink /etc/systemd/system/multi-
user.target.wants/nagios.service →
/usr/lib/systemd/system/nagios.service.

*** Init script installed ***
```

6) Installer et configurer les permissions du fichier des commandes externes.

```
[root@localhost nagios-4.5.2]# 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 ***
```

7) Installer les fichiers de configuration dans /usr/local/nagios/etc.

```
[root@localhost nagios-4.5.2]# make install-config

/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d
/usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg
/usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg
/usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg
/usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

Remember, these are *SAMPLE* config files.  You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.
```

8) Installer le fichier de configuration Apache de l'interface web de Nagios.

```
[root@localhost nagios-4.5.2]# make install-webconf

/usr/bin/install -c -m 644 sample-config/httpd.conf
/etc/httpd/conf.d/nagios.conf
if [ 0 -eq 1 ]; then \
        ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-
enabled/nagios.conf; \
fi

*** Nagios/Apache conf file installed ***
```

9) Installer le nouveau thème Exfoliation de l'interface web de Nagios.

```
[root@localhost nagios-4.5.2]# make install-exfoliation

*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme
```

10) Installer le thème classique de l'interface web de Nagios (ne pas faire cette étape si on choisit le nouveau thème)

```
[root@localhost nagios-4.5.2]# make install-classicui

*** Classic theme installed ***
NOTE: Use 'make install-exfoliation' to use new Nagios theme
```

6 Créer l'utilisateur de l'interface web Nagios

Ajouter un utilisateur pour accéder à l'interface Web de Nagios.

```
[root@localhost nagios-4.5.2]# htpasswd -c
/usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
```

Redémarrer le service apache pour activer les changements.

```
[root@localhost nagios-4.5.2]# systemctl restart httpd
```

Revenir au répertoire personnel.

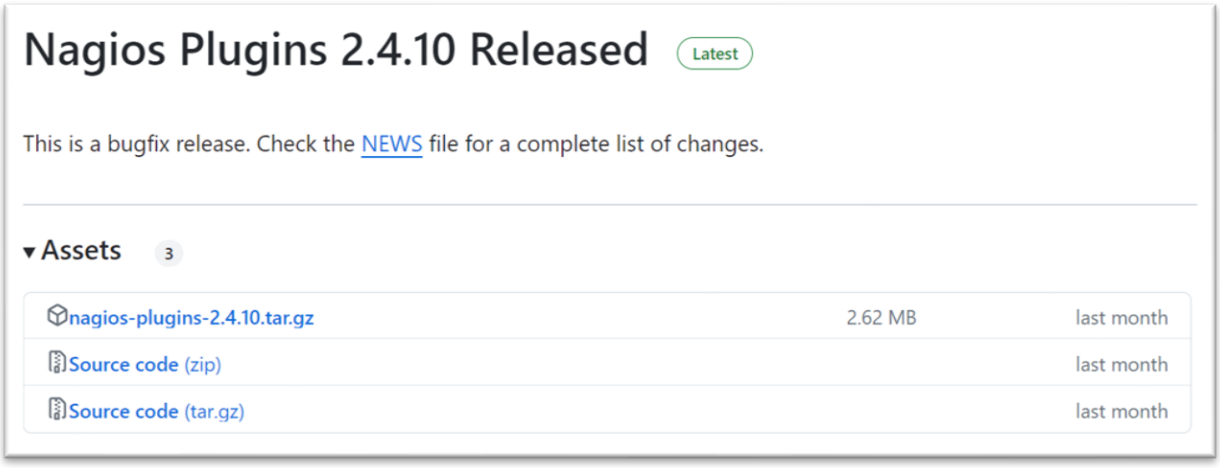
```
[root@localhost nagios-4.5.2]# cd
[root@localhost ~]#
```

7 Installer les Plugins Nagios

Les plugins Nagios permettent d'étendre les fonctionnalités de monitorage de Nagios.

Consulter la page des versions de Github pour identifier la dernière version disponible des plugins Nagios.

https://github.com/nagios-plugins/nagios-plugins/releases




```
[root@localhost ~]# export VER="2.4.10"
```

Télécharger les plugins Nagios.

```
[root@localhost ~]# curl -SL https://github.com/nagios-plugins/nagios-plugins/releases/download/release-$VER/nagios-plugins-$VER.tar.gz | tar -xzf -

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 2680k  100 2680k    0     0  1962k      0  0:00:01  0:00:01 --:--:-- 9348k
```

Se déplacer sur le répertoire source des plugins Nagios.

```
[root@localhost ~]# cd nagios-plugins-$VER
[root@localhost nagios-plugins-2.4.10]#
```

Exécuter le script configure.

```
[root@localhost nagios-plugins-2.4.10]# ./configure --with-nagios-user=nagios --with-nagios-group=nagios
```

Compiler les plugins Nagios.

```
[root@localhost nagios-plugins-2.4.10]# make
```

Installer les plugins Nagios.

```
[root@localhost nagios-plugins-2.4.10]# make install
```

Revenir au répertoire personnel.

```
[root@localhost nagios-plugins-2.4.10]# cd
[root@localhost ~]#
```

8 Valider l'installation de Nagios

Valider l'installation de Nagios.

```
[root@localhost ~]# /usr/local/nagios/bin/nagios -v
/usr/local/nagios/etc/nagios.cfg

Nagios Core 4.5.2
Copyright (c) 2009-present Nagios Core Development Team and
Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-04-30
License: GPL

Website: https://www.nagios.org
Reading configuration data...
   Read main config file okay...
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 8 services.
        Checked 1 hosts.
        Checked 1 host groups.
        Checked 0 service groups.
        Checked 1 contacts.
        Checked 1 contact groups.
        Checked 24 commands.
        Checked 5 time periods.
        Checked 0 host escalations.
        Checked 0 service escalations.
Checking for circular paths...
        Checked 1 hosts
        Checked 0 service dependencies
        Checked 0 host dependencies
        Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors:   0

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

9 Activer et démarrer le service Nagios

Activer le démarrer le service Nagios.

```
[root@localhost ~]# systemctl  enable nagios --now
```

Vérifier le statut du service Nagios (running).

```
[root@localhost ~]# systemctl status nagios

● nagios.service - Nagios Core 4.5.2
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset:
disabled)
     Active: active (running) since Tue 2024-05-28 20:49:19 EDT; 9s ago
       Docs: https://www.nagios.org/documentation
    Process: 69211 ExecStartPre=/usr/local/nagios/bin/nagios -v
/usr/local/nagios/etc/nag>
    Process: 69212 ExecStart=/usr/local/nagios/bin/nagios -d
/usr/local/nagios/etc/nagios>
   Main PID: 69213 (nagios)
      Tasks: 6 (limit: 10899)
     Memory: 5.9M
        CPU: 194ms
     CGroup: /system.slice/nagios.service
             ├─69213 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─69214 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio>
             ├─69215 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio>
             ├─69216 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio>
             ├─69217 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio>
             └─69218 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

May 28 20:49:19 localhost nagios[69213]: qh: Socket
'/usr/local/nagios/var/rw/nagios.qh' >
May 28 20:49:19 localhost nagios[69213]: qh: core query handler registered
May 28 20:49:19 localhost nagios[69213]: qh: echo service query handler registered
May 28 20:49:19 localhost nagios[69213]: qh: help for the query handler registered
May 28 20:49:19 localhost nagios[69213]: wproc: Successfully registered manager as
```

10 Autoriser l'accès à l'interface web de Nagios

Autoriser les protocole http et https sur le firewall (si le firewall est activé).

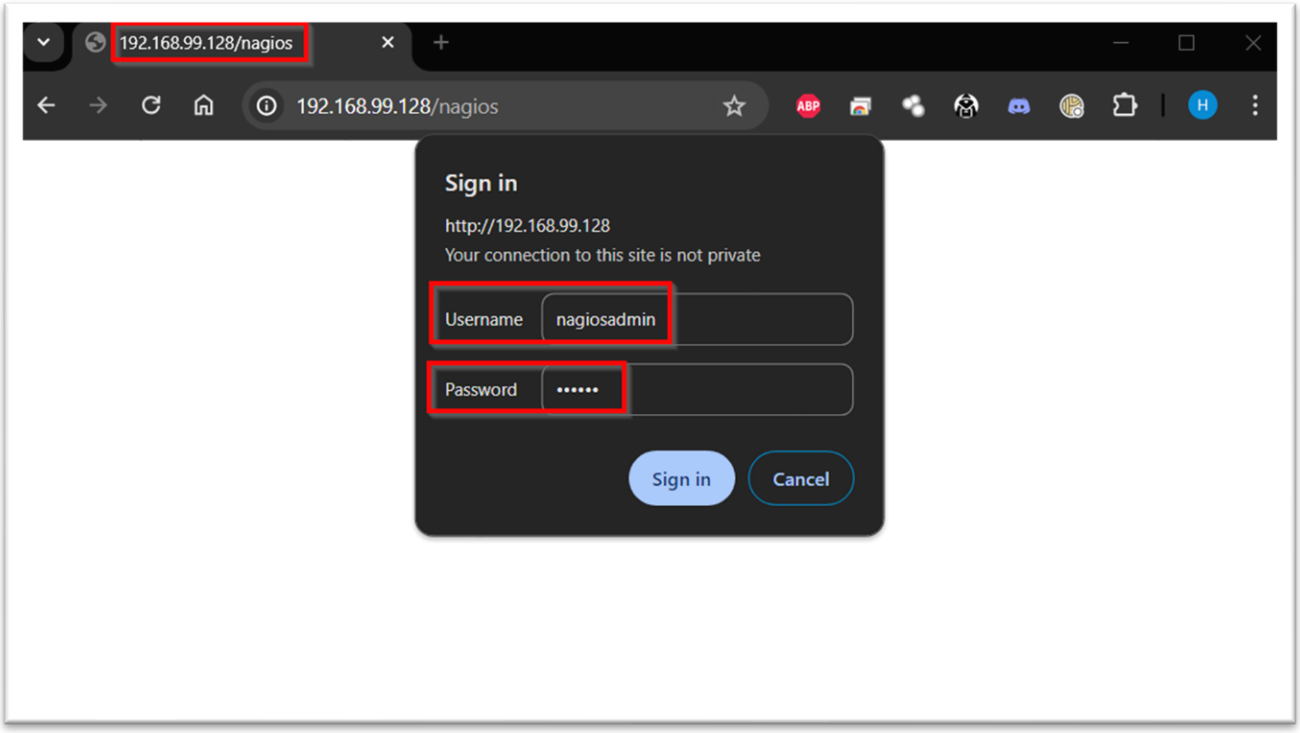
```
[root@localhost ~]# firewall-cmd --permanent  --add-service={http,https}

Success
```

```
[root@localhost ~]# firewall-cmd –reload

success
```

11 Accéder à l'interface web de Nagios

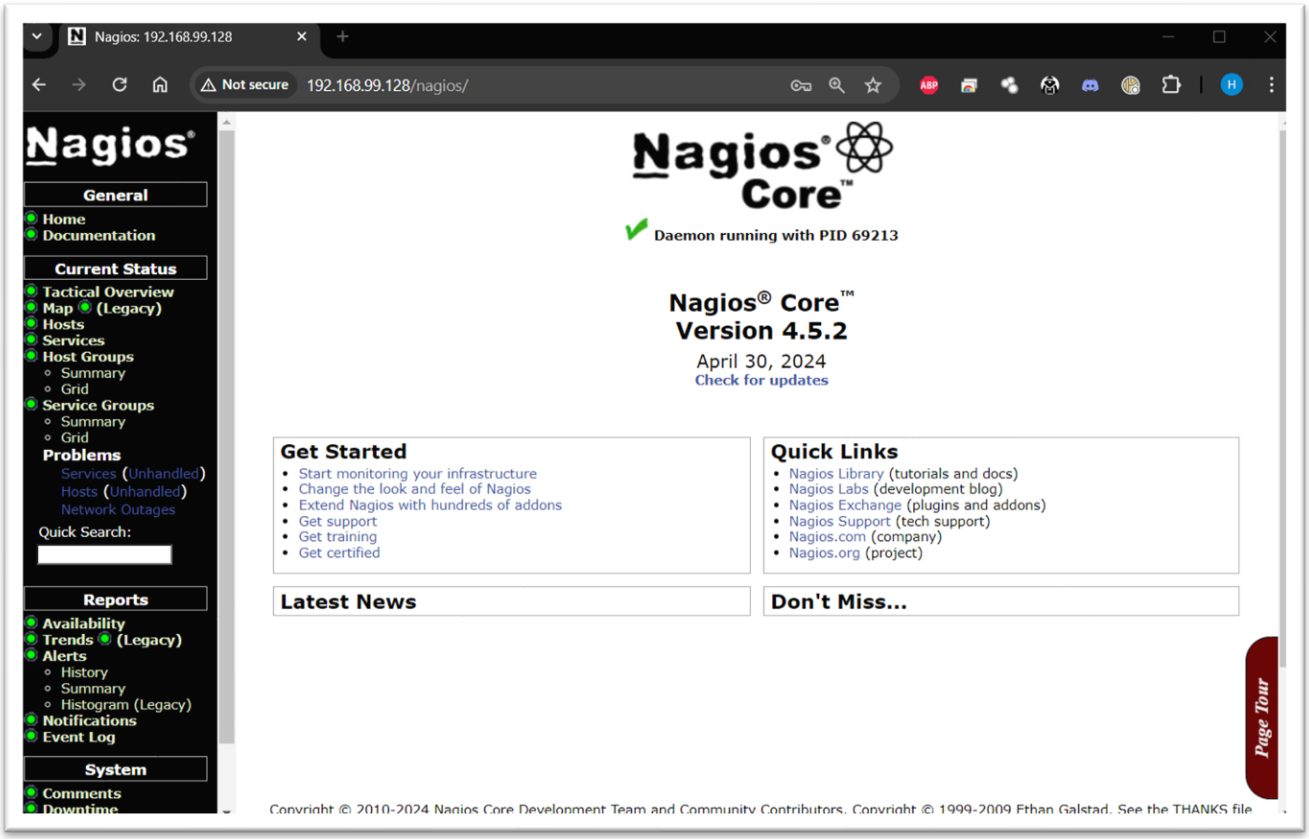
Accéder à l'interface web avec l'url :

http://adresse_ip/nagios



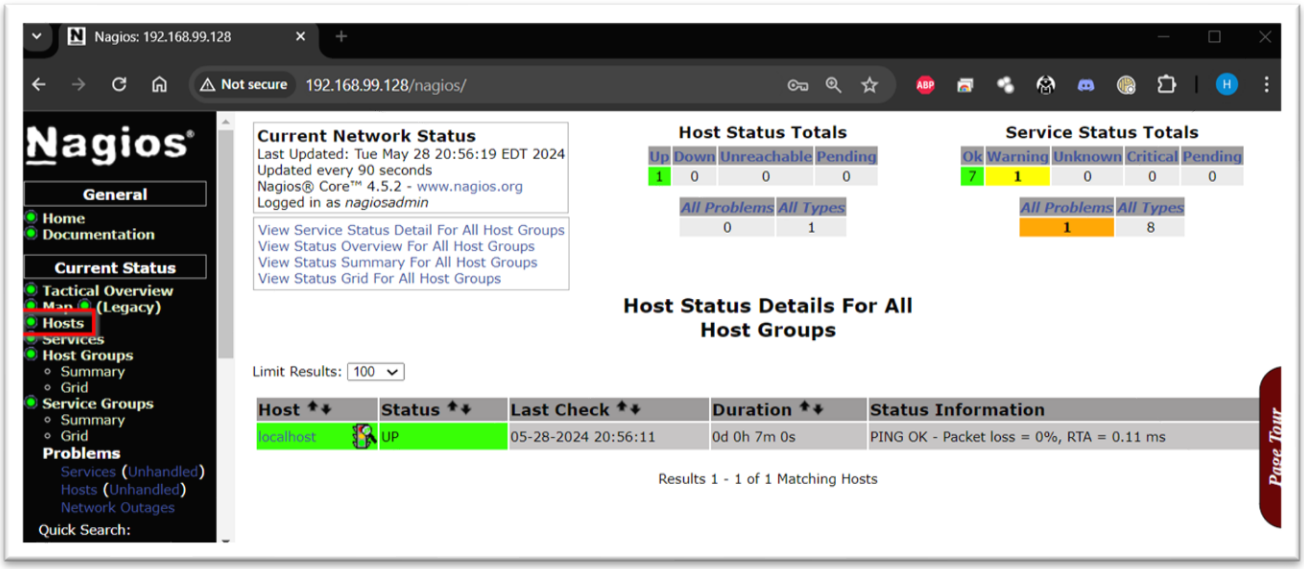
S'authentifier avec l'utilisateur et le mot de nagios.

L’interface web de Nagios s’affichera :



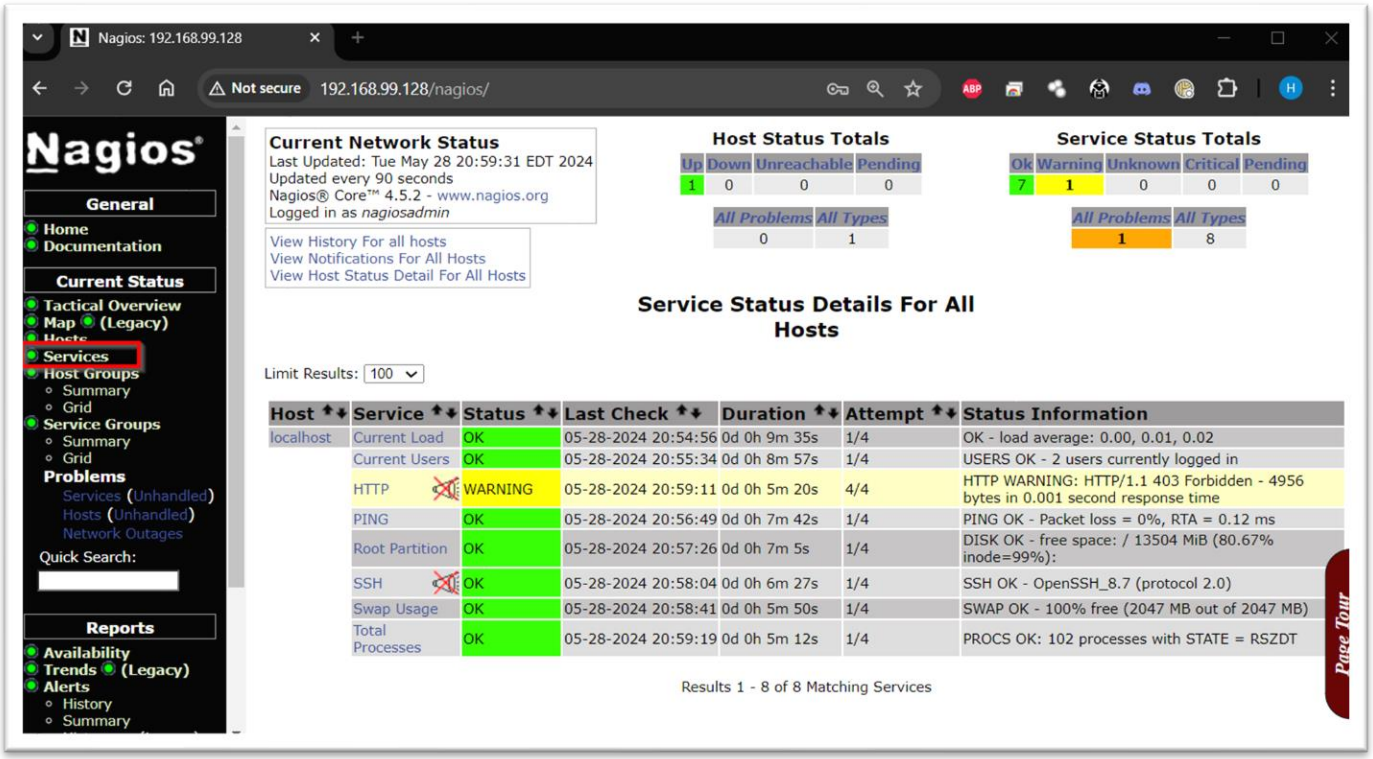
La dernière version de Nagios est maintenant installée. Il reste à configurer les cibles de surveillance Nagios.

Afficher les hôtes surveillés par Nagios



On un seul hôte surveillé (locahost). C’est le serveur Nagios lui-même.

Afficher les services surveillés par Nagios



On remarque que le service http a le statut WARNING. Car on n'a aucun document web dans le répertoire /var/www/html.

Il suffit de créer un document web pour avoir un statut OK :

```
[root@localhost ~]# touch /var/www/html/index.html
```

Le statut du service http change pour OK.