JUDE CHRYST DEGILLES
POUMINA MABONZO
Bac+3 - AIS - 2025
judechrystpoums@gmail.com

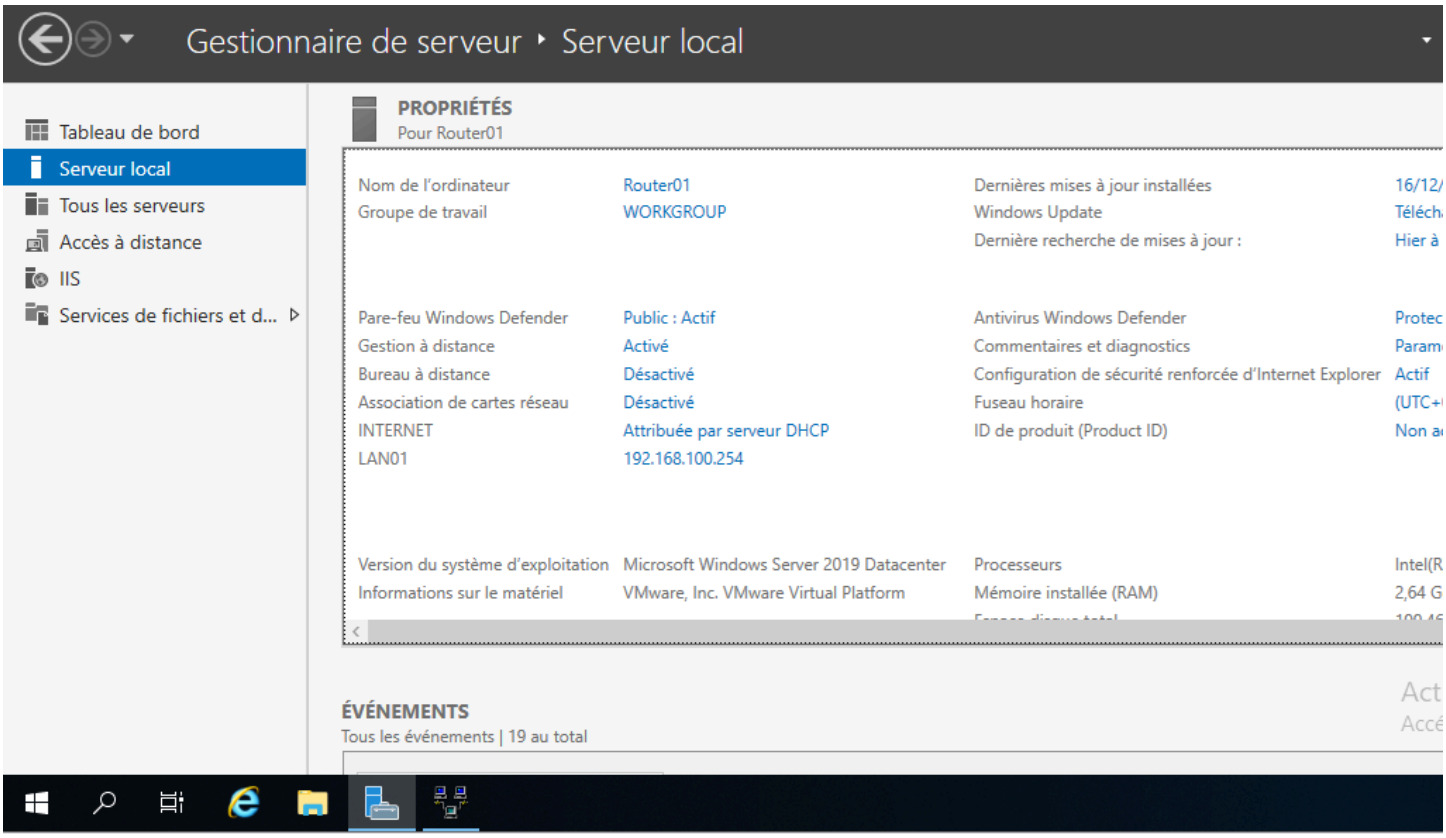
Nagios®

METTRE EN PLACE UN SERVEUR DE SUPERVISION

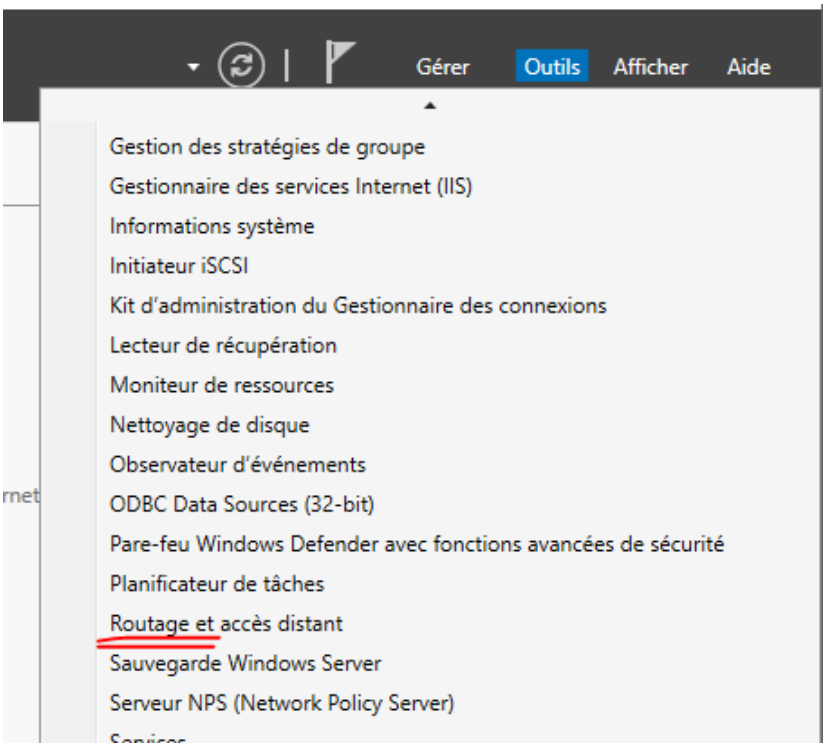
Machine Active directory Machine Routeur	Machine nagios

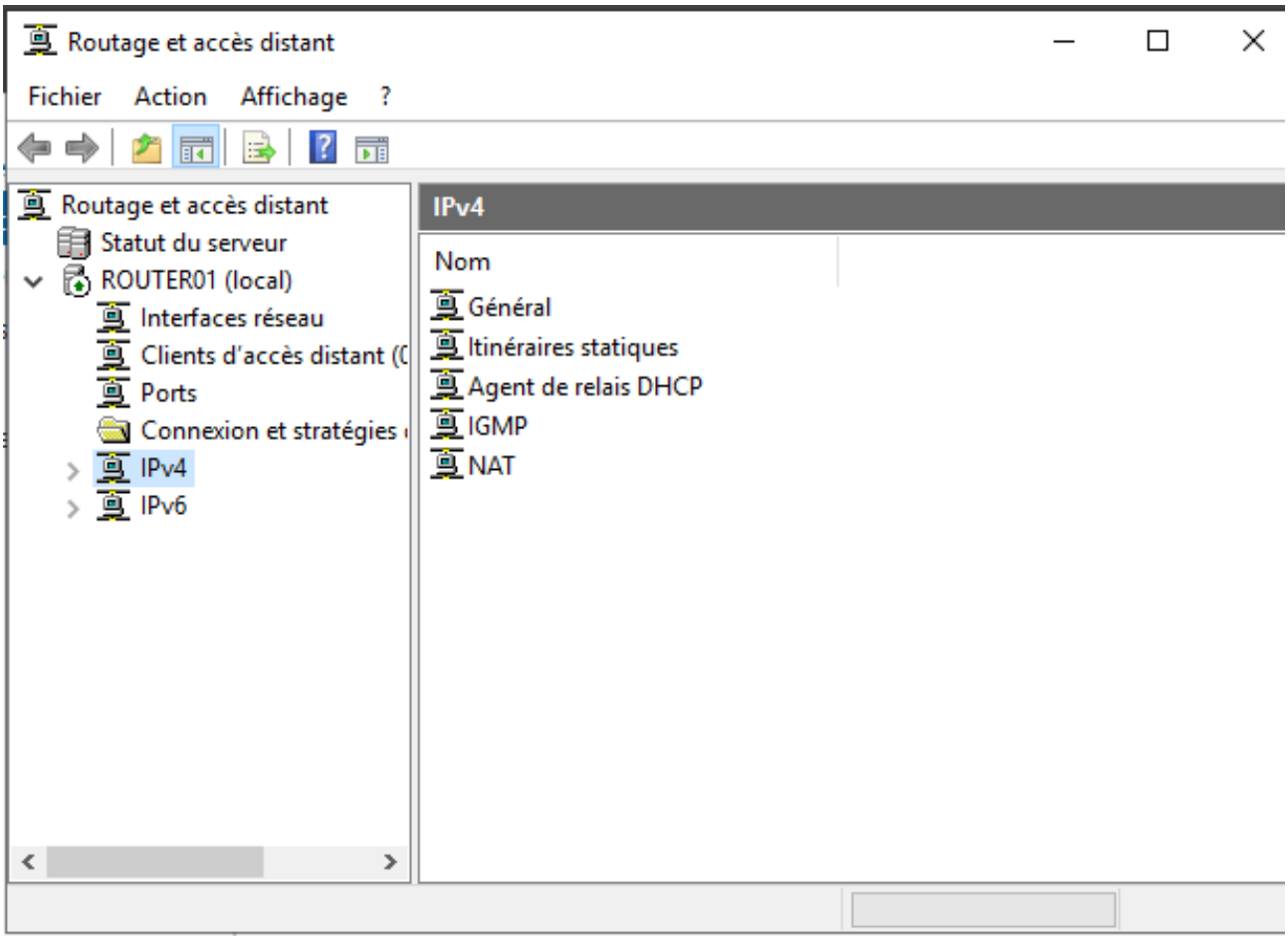
Lancer la machine Windows server qui joue le rôle de routeur

1. Démarrer votre **routeur** (dédier Windows serveur 2019)
2. Mettre quelque prérequis comme : Nom de l'ordinateur, avoir deux cartes réseaux INTERNET et LAN avec leur adresse IP
3. Ajouter le rôle Accès à distance et routage en cliquant sur gérer et nouveau rôle



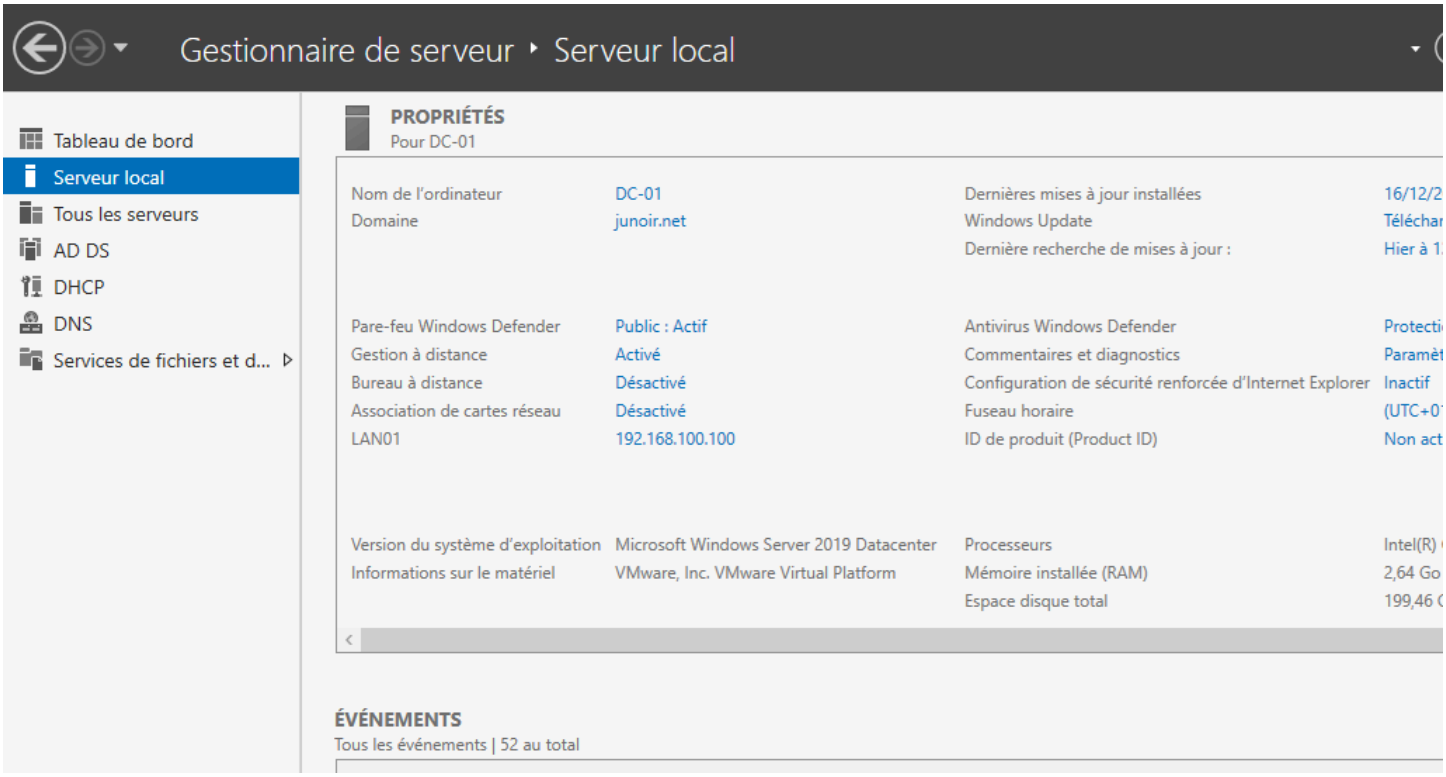
1. Apres avoir ajouter le rôle accès à distance aller dans **Outils / Routage et accès à distance**
2. on passe à la configuration et le faire clic droit sur le nom de votre routeur **(configurer et activer le routage)**
3. Choisir Accès VPN



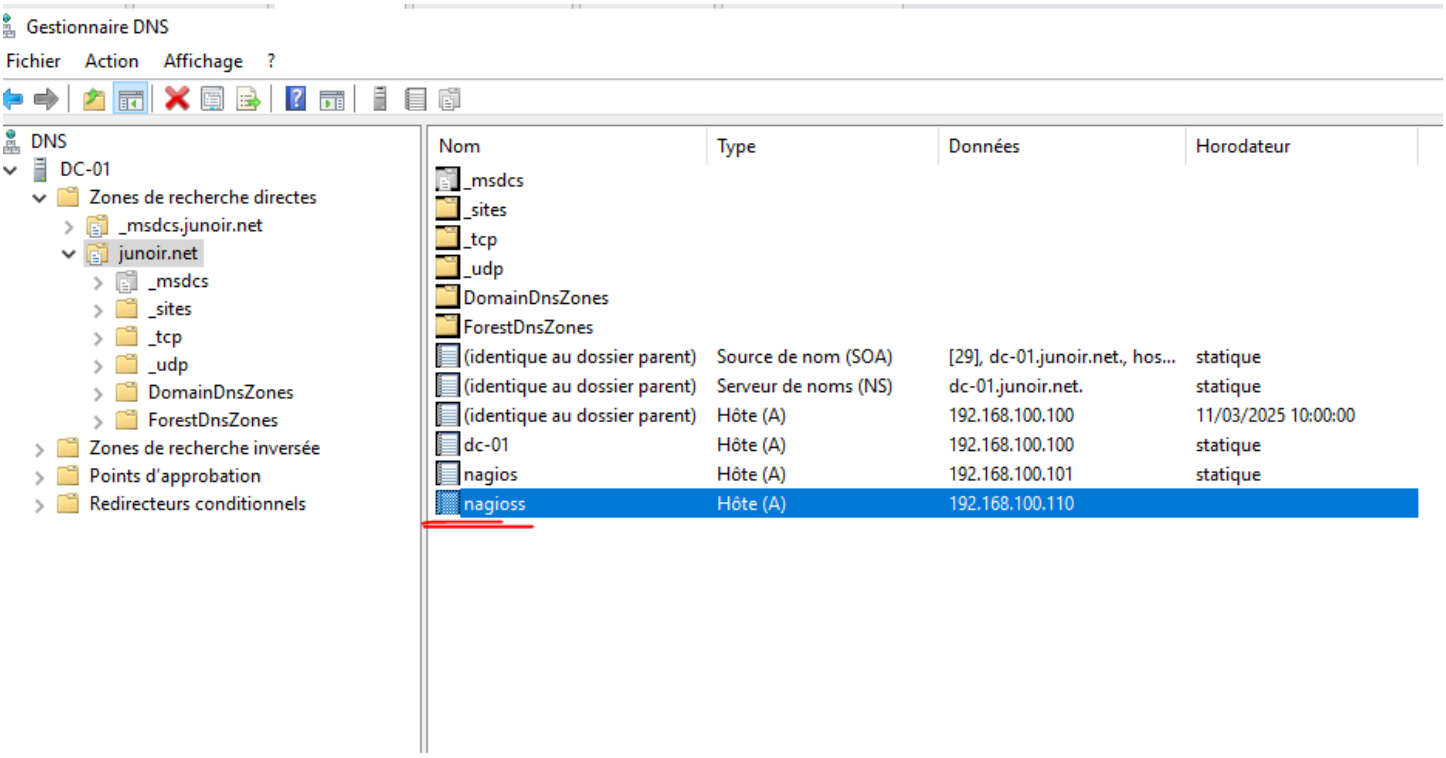


Lancer la machine Windows server qui joue le rôle de Active Directory

1. Démarrer votre **DC-01** (dédier Windows serveur 2019)
2. Mettre quelque prérequis comme : Nom de l'ordinateur, avoir une carte réseau LAN avec son adresse IP
3. Ajouter les rôle **AD DS, DHCP** en cliquant sur gérer et nouveau rôle

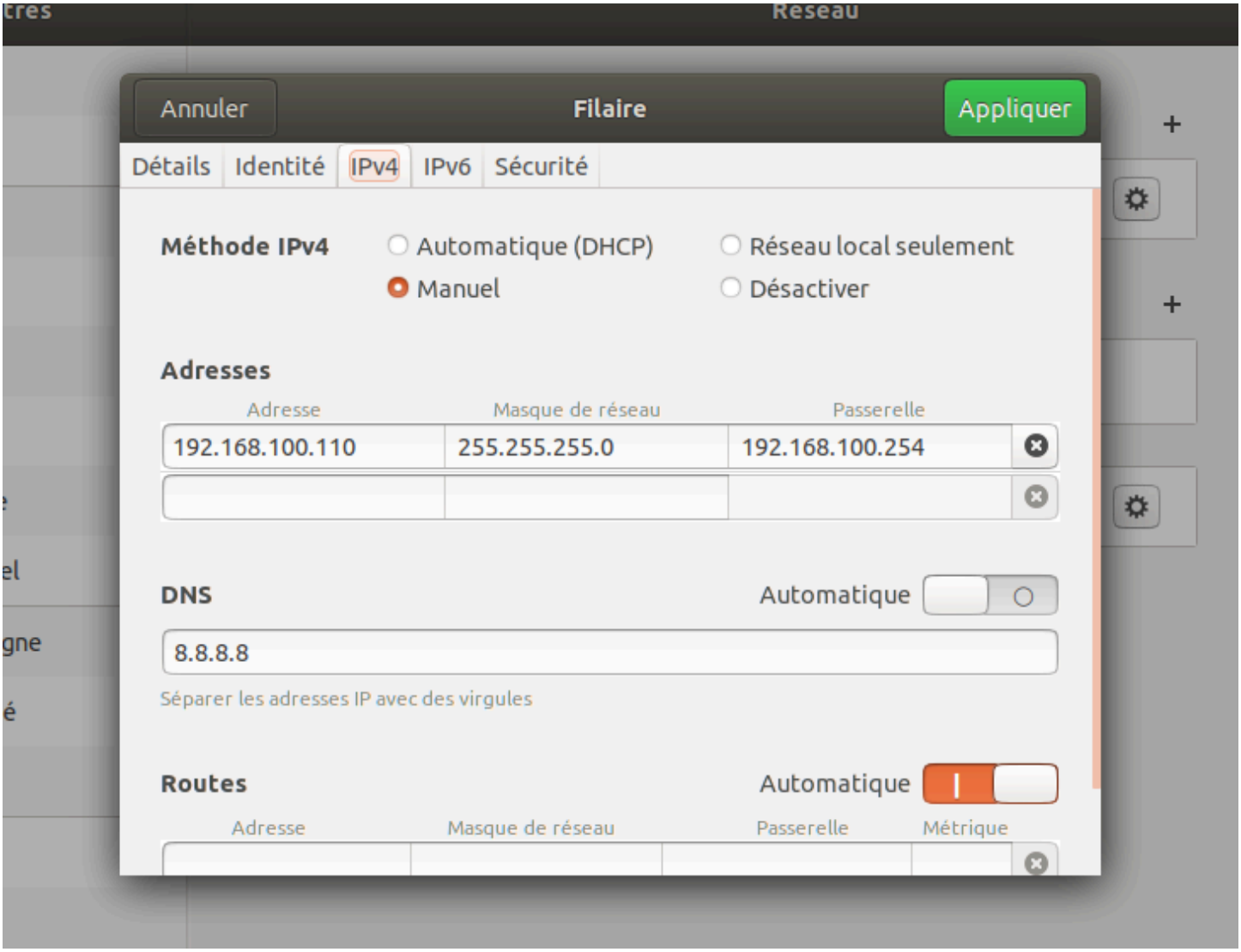


1. Configurer le **DNS** et le **DHCP**



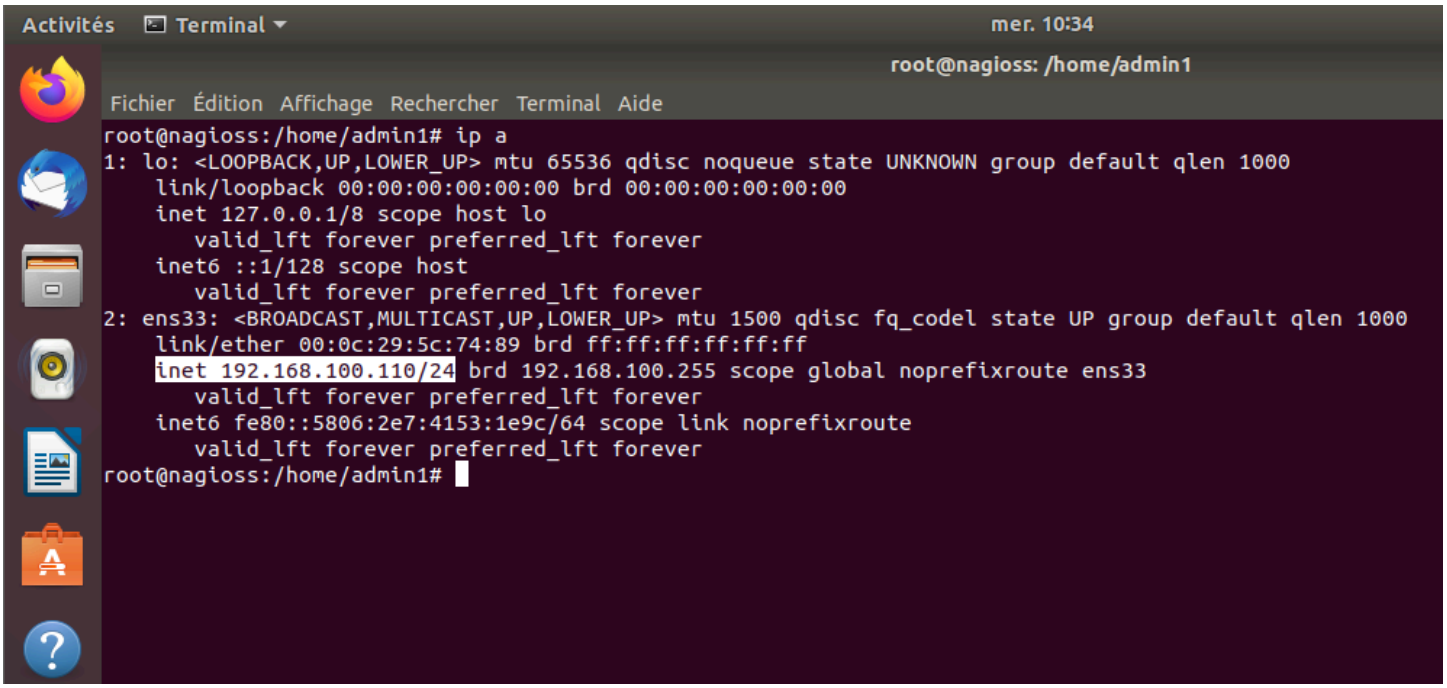
Lancer Ubuntu Linux qui joue le rôle de serveur Nagios1

1. Démarrer votre **machine nagios1** (dédier Linux Ubuntu 18.0.4)
2. Mettre votre machine nagios1 dans une interface LAN ensuite lui attribuer une IP statique. La passerelle est diriger vers le Routeur et saisir IP de Google



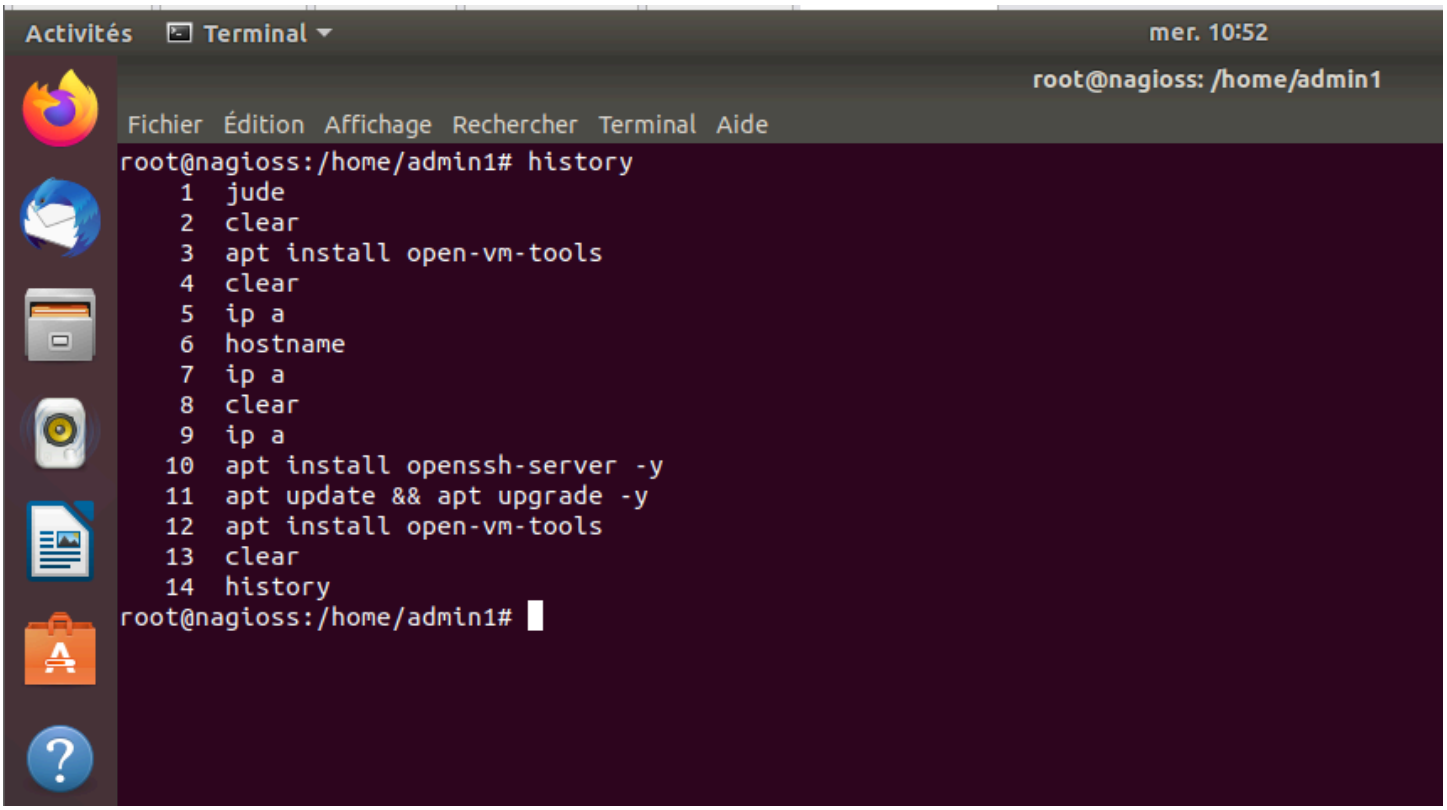
1. Vérifier si l'attribuer dans IP a marcher avec la commande:

- **ip a**



1. Avant d'installer Nagios, assurez-vous que votre système est à jour et dispose des paquets suivants :

- **sudo apt update && sudo apt upgrade -y**
- **sudo apt install openssh-server -y**
- **sudo apt install open-vm-tools**



- **sudo apt install -y apache2 php libapache2-mod-php build-essential unzip curl openssl libssl-dev wget bc gawk dc libmcrypt-dev libssl-dev autoconf gcc libc6 make libgd-dev**

```
Creating config file /etc/php/7.2/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php7.2
Paramétrage de libfontconfig1-dev:amd64 (2.12.6-0ubuntu2) ...
Paramétrage de libtiff-dev (4.0.9-5ubuntu0.10) ...
Paramétrage de libapache2-mod-php (1:7.2+60ubuntu1) ...
Paramétrage de build-essential (12.4ubuntu1) ...
Paramétrage de libgd-dev:amd64 (2.2.5-4ubuntu0.5) ...
Paramétrage de php7.2 (7.2.24-0ubuntu0.18.04.17) ...
Paramétrage de php (1:7.2+60ubuntu1) ...
Traitement des actions différées (« triggers ») pour systemd (237-3ubuntu10.57) ..
Traitement des actions différées (« triggers ») pour man-db (2.8.3-2ubuntu0.1) ...
Traitement des actions différées (« triggers ») pour ufw (0.36-0ubuntu0.18.04.2) .
Traitement des actions différées (« triggers ») pour ureadahead (0.100.0-21) ...
Traitement des actions différées (« triggers ») pour install-info (6.5.0.dfsg.1-2)
Traitement des actions différées (« triggers ») pour libc-bin (2.27-3ubuntu1.6) ..
root@nagioss:/home/admin1# sudo apt install -y apache2 php libapache2-mod-php buil
d-essential unzip curl     openssl libssl-dev wget bc gawk dc libmcrypt-dev libssl
-dev     autoconf gcc libc6 make libgd-dev
```

1. Faire un test de ping vers le server AD : **ping 192.168.100.100**

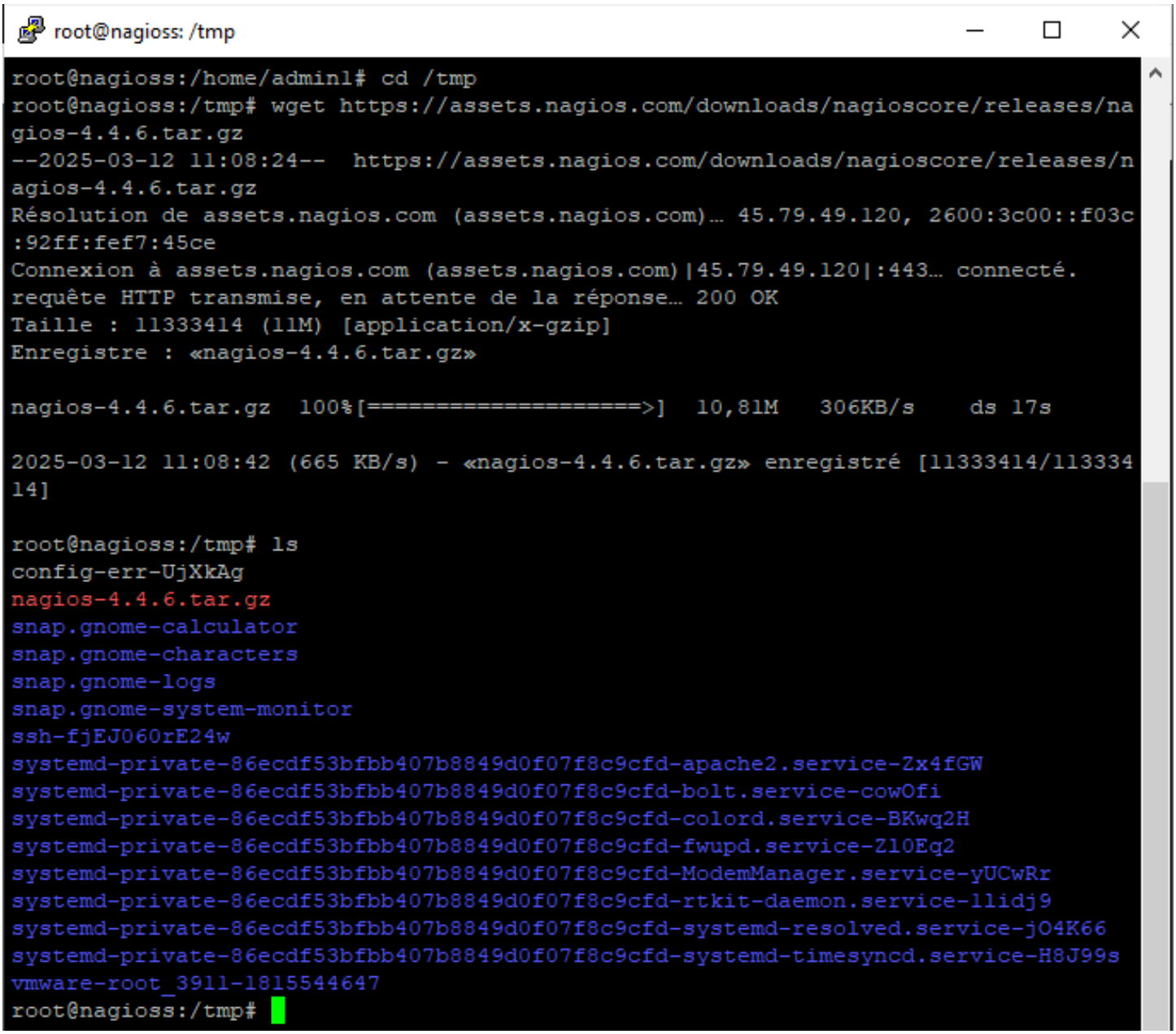
- **sudo useradd nagios**
- **sudo groupadd nagcmd**
- **sudo usermod -aG nagcmd nagios**
- **sudo usermod -aG nagcmd www-data**

root@nagioss: /home/admin1

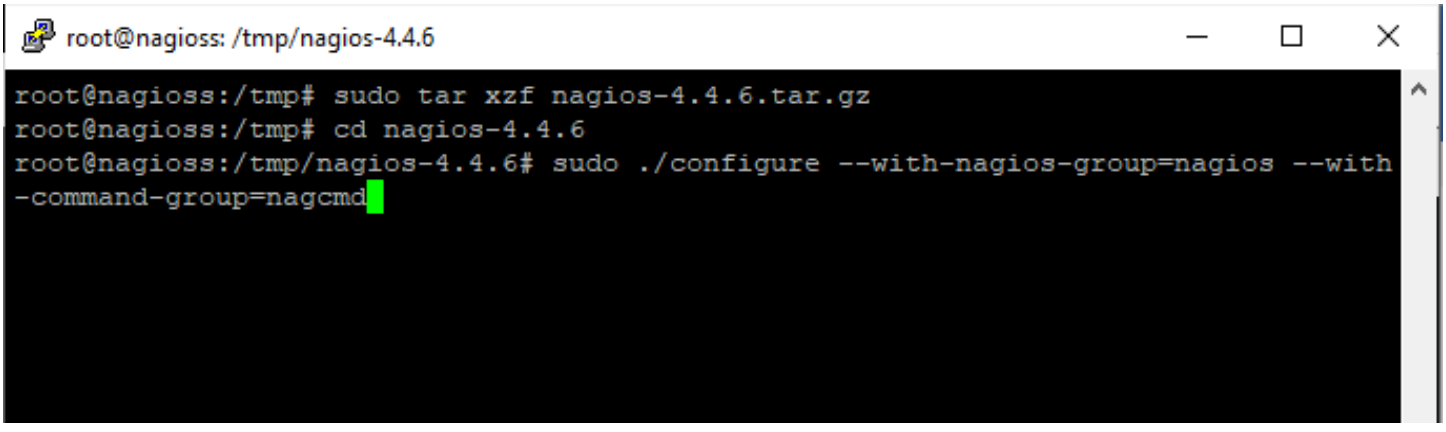
```
root@nagioss:/home/admin1# ping 192.168.100.100
PING 192.168.100.100 (192.168.100.100) 56(84) bytes of data.
64 bytes from 192.168.100.100: icmp_seq=1 ttl=128 time=1.07 ms
64 bytes from 192.168.100.100: icmp_seq=2 ttl=128 time=1.41 ms
64 bytes from 192.168.100.100: icmp_seq=3 ttl=128 time=2.59 ms
^C
--- 192.168.100.100 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 1.079/1.696/2.592/0.649 ms
root@nagioss:/home/admin1# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=127 time=8.91 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=127 time=14.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=127 time=8.69 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 8.699/10.695/14.474/2.675 ms
root@nagioss:/home/admin1# sudo useradd nagios
root@nagioss:/home/admin1# sudo groupadd nagcmd
root@nagioss:/home/admin1# sudo usermod -aG nagcmd nagios
root@nagioss:/home/admin1# sudo usermod -aG nagcmd www-data
root@nagioss:/home/admin1#
```

1. Téléchargez et installez Nagios core

- **cd /tmp**
- **wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz**
- **sudo tar xzf nagios-4.4.6.tar.gz**



- **cd nagios-4.4.6**
- **sudo ./configure --with-nagios-group=nagios --with-command-group=nagcmd**



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


*** Configuration summary for nagios 4.4.6 2020-04-28 ***:

 General Options:
 -------------------------
        Nagios executable:  nagios
        Nagios user/group:  nagios,nagios
       Command user/group:  nagios,nagcmd
            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/apache2/sites-available
             Mail program:  /bin/mail
                  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.

root@nagioss:/tmp/nagios-4.4.6#
```

```
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@nagioss:/tmp/nagios-4.4.6# sudo make all
```

- **sudo make all**

- **sudo make install**
- **sudo make install-init**
- **sudo make install-daemoninit**
- **sudo make install-config**

```
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] : on quitte le répertoire « /tmp/nagios-4.4.6 »
root@nagioss:/tmp/nagios-4.4.6# sudo make install
```

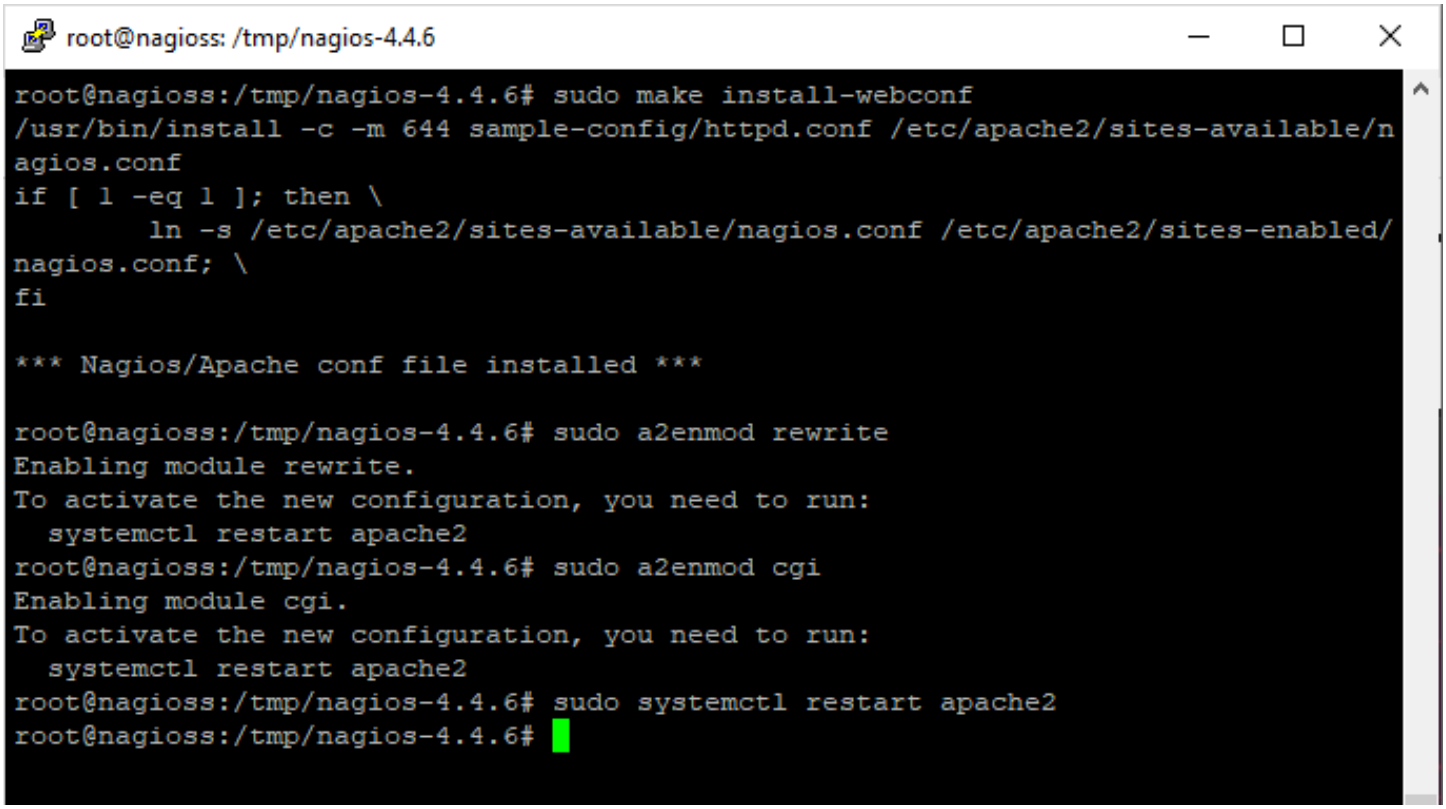
```
root@nagioss:/tmp/nagios-4.4.6# sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/sy
stem/nagios.service
root@nagioss:/tmp/nagios-4.4.6# sudo make install-daemoninit
/usr/bin/install -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/sy
stem/nagios.service
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /lib/
systemd/system/nagios.service.

*** Init script installed ***

root@nagioss:/tmp/nagios-4.4.6# sudo make install-config
```

- **sudo make install-commandmode**

```
*** 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.

root@nagioss:/tmp/nagios-4.4.6# sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

root@nagioss:/tmp/nagios-4.4.6#
```

1. Installez et configurez l'interface web

- **sudo make install-webconf**
- **sudo a2enmod rewrite**
- **sudo a2enmod cgi**
- **sudo systemctl restart apache2**

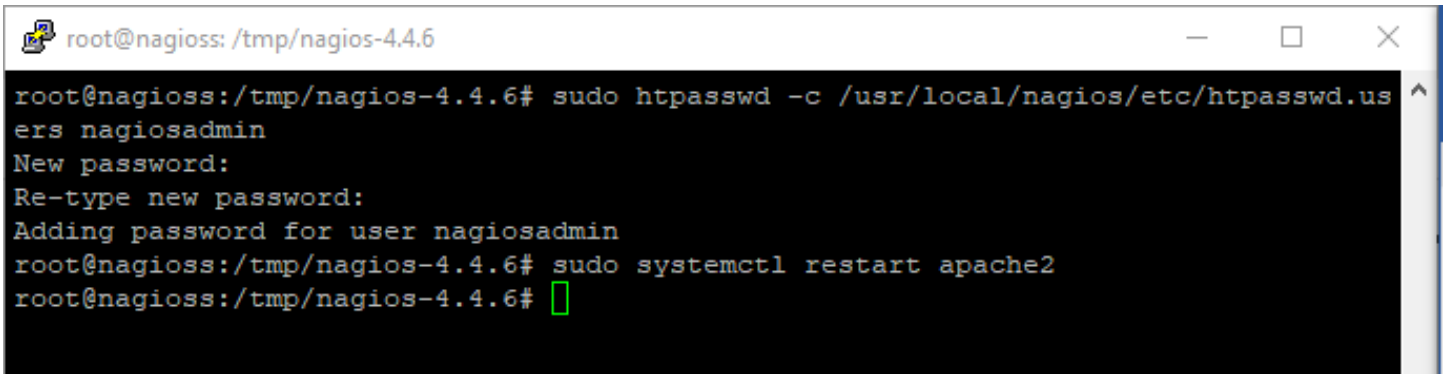


1. Créer un utilisateur pour accéder à l'interface nagios

- **sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin**

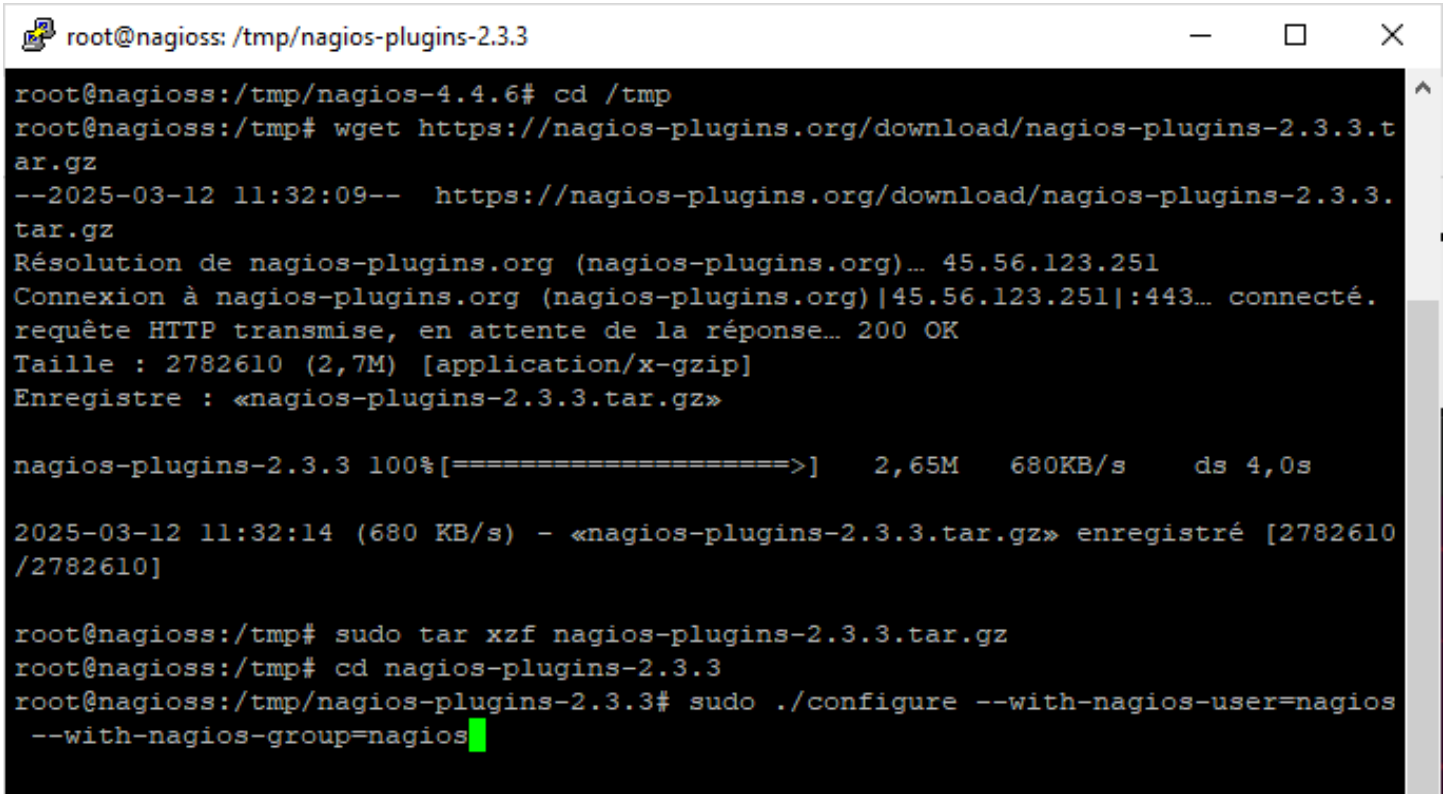
Entrer un mot de passe par exemple **: P@ssword**

- **sudo systemctl restart apache2**

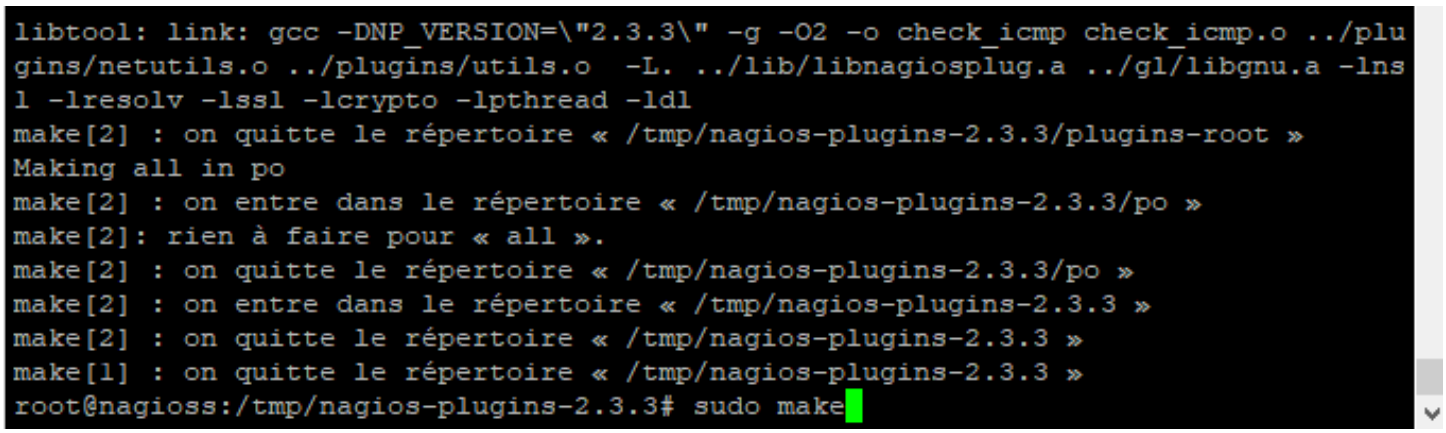


1. Installation des plugins nagios

- **cd /tmp**
- **wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz**
- **sudo tar xzf nagios-plugins-2.3.3.tar.gz**
- **cd nagios-plugins-2.3.3**
- **sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios**

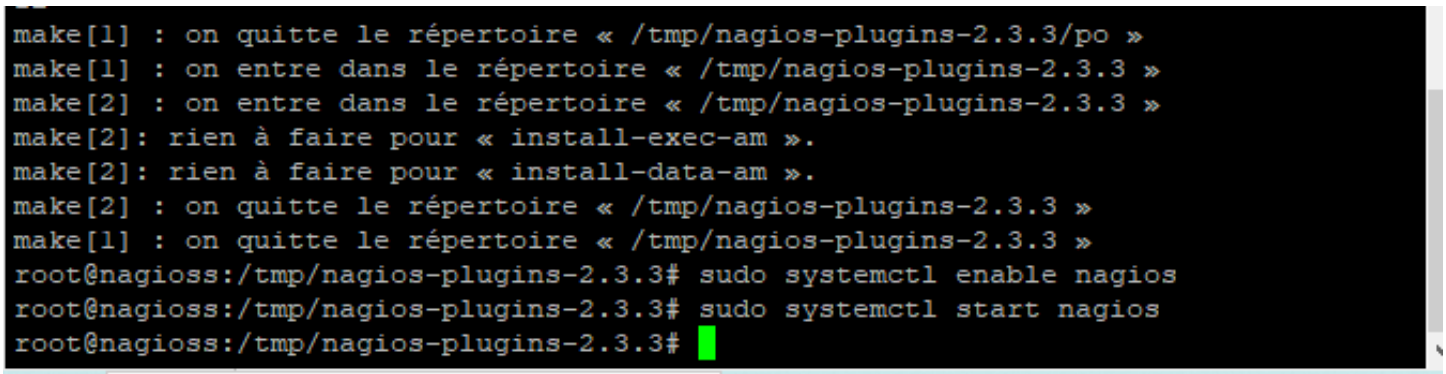


- **sudo make**
- **sudo make install**



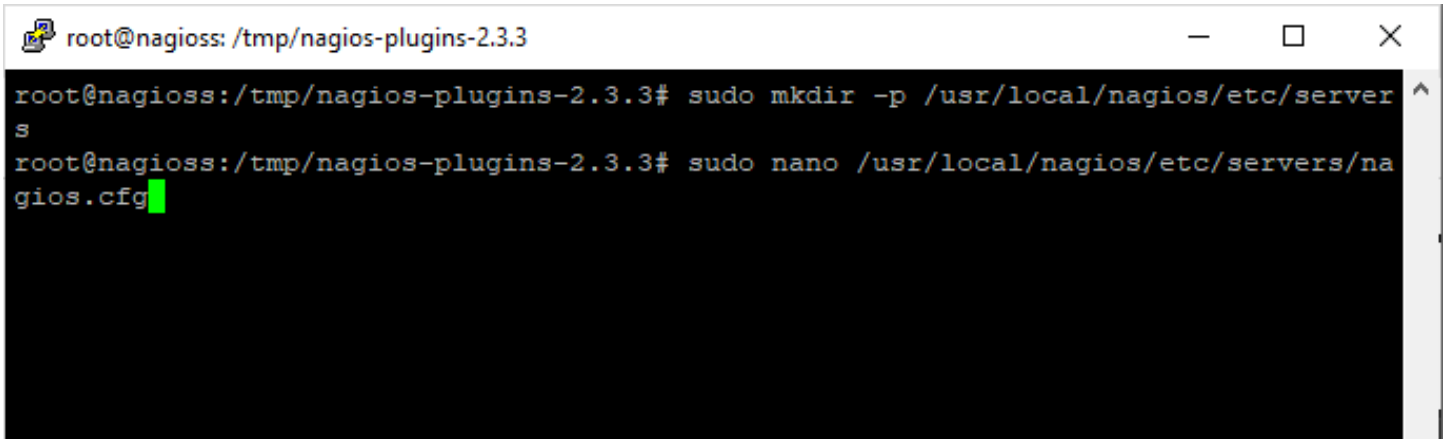
1. Démarrez nagios

- **sudo systemctl enable nagios**
- **sudo systemctl start nagios**



1. Ajoutez un hôte à surveiller
2. Si ce dossier n'existe pas, crée-le :

- **sudo mkdir -p /usr/local/nagios/etc/servers**
- **sudo nano /usr/local/nagios/etc/servers/nagios.cfg**



Ajoute cette configuration :

define host {

use linux-server

host_name nagios1

alias Serveur Nagios

address 192.168.100.110

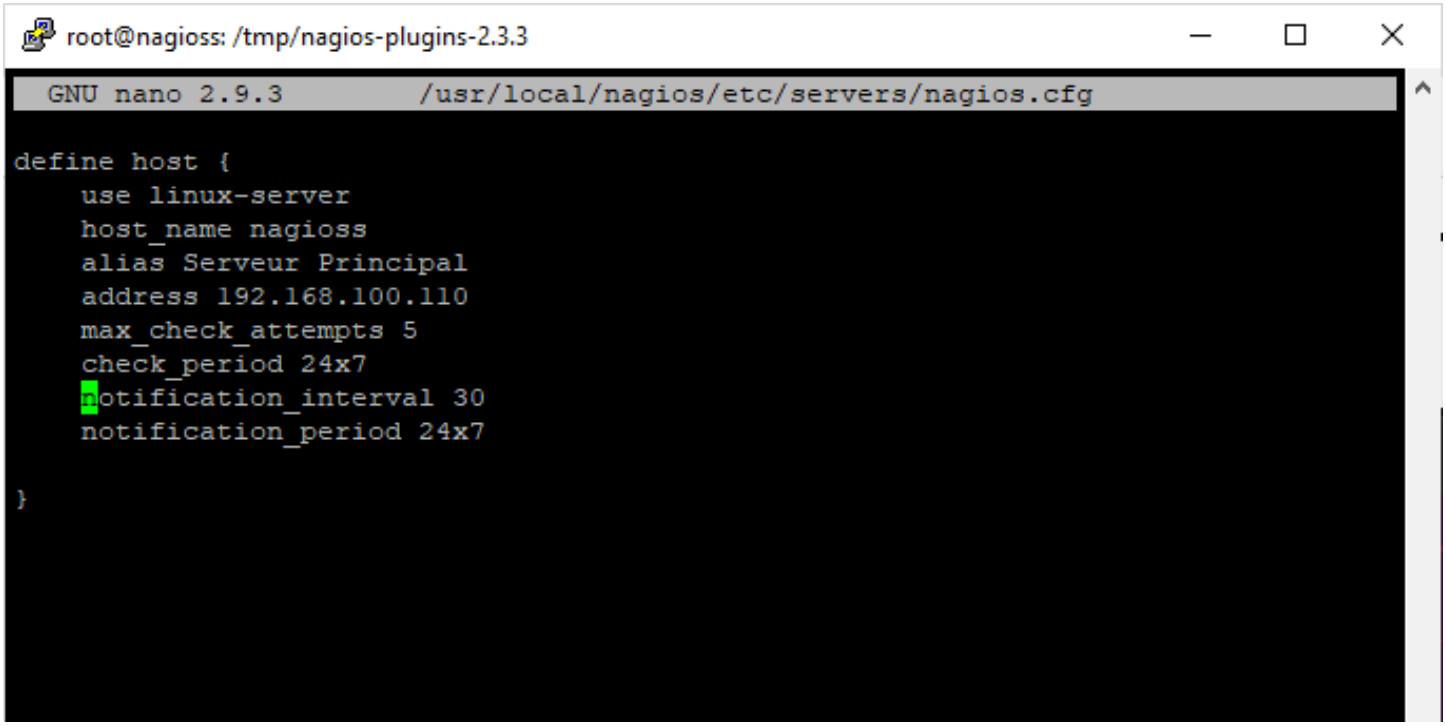
max_check_attempts 5

check_period 24x7

notification_interval 30

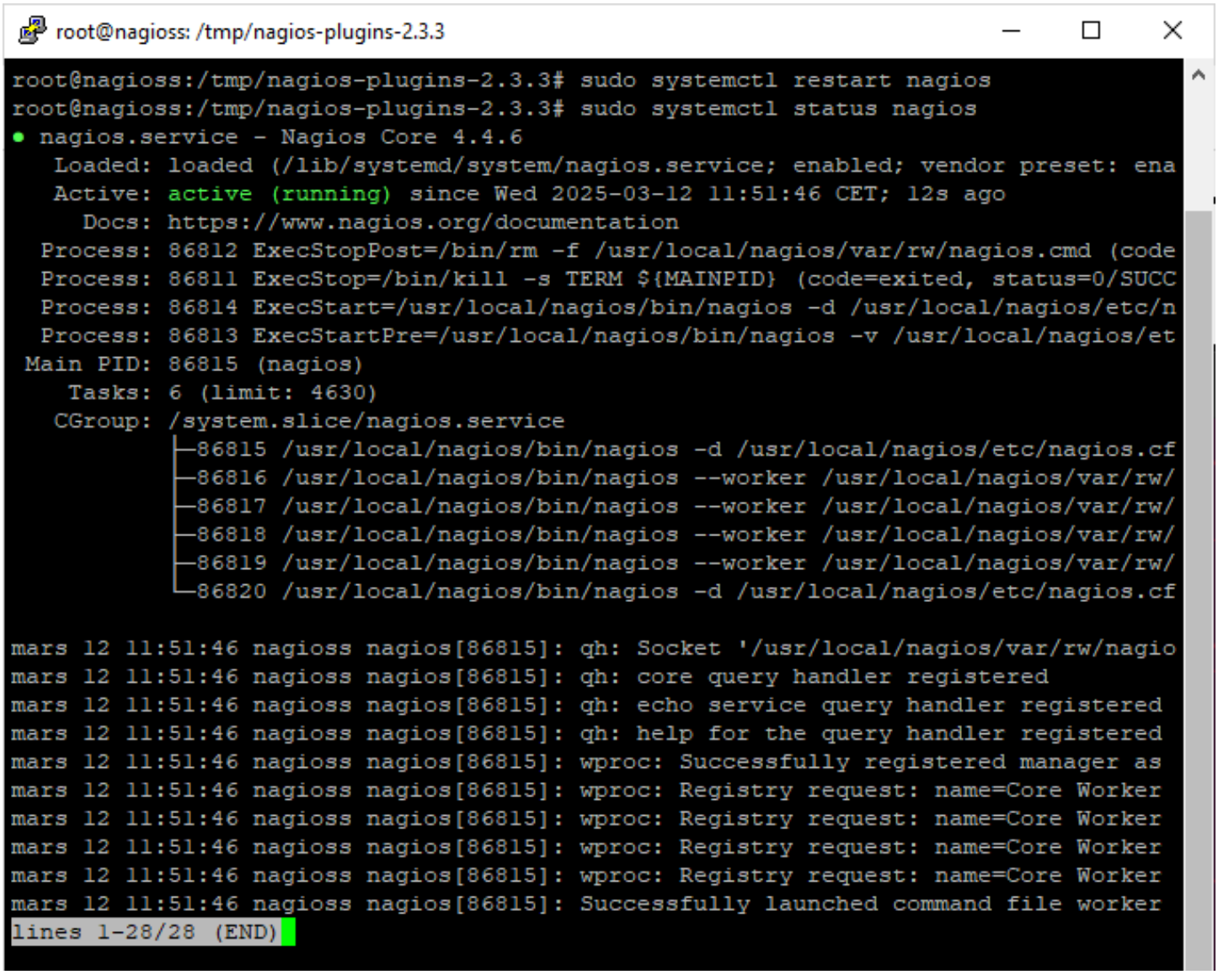
notification_period 24x7

}



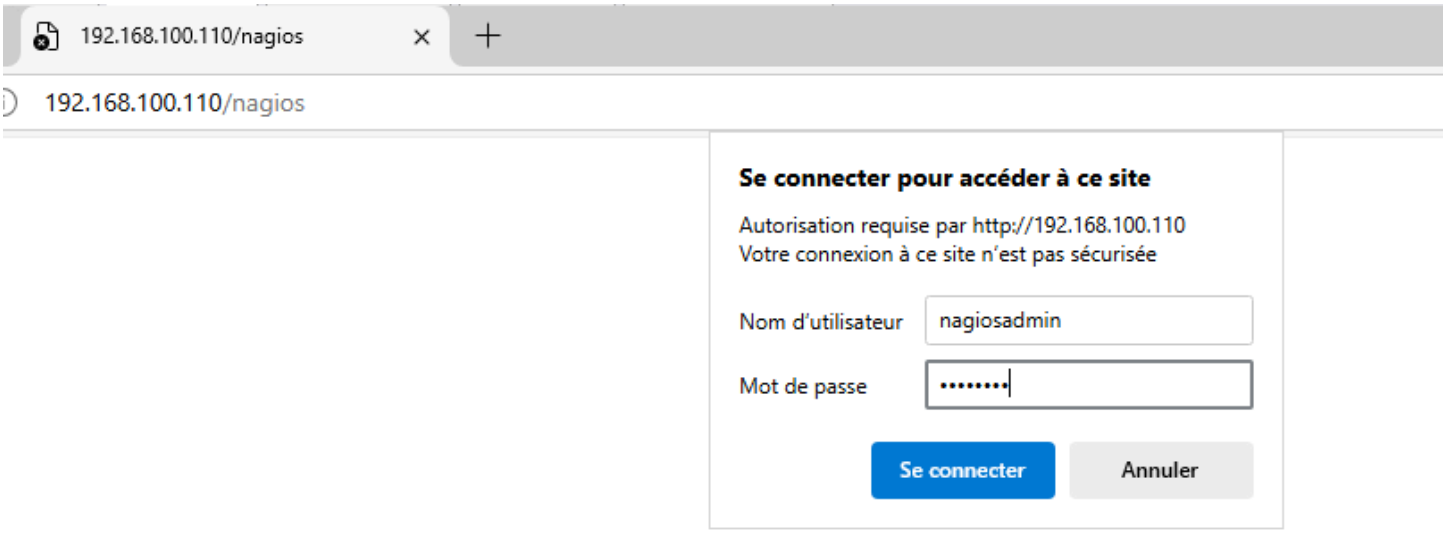
1. Redémarrez Nagios :

- **sudo systemctl restart nagios**
- **sudo systemctl status nagios**

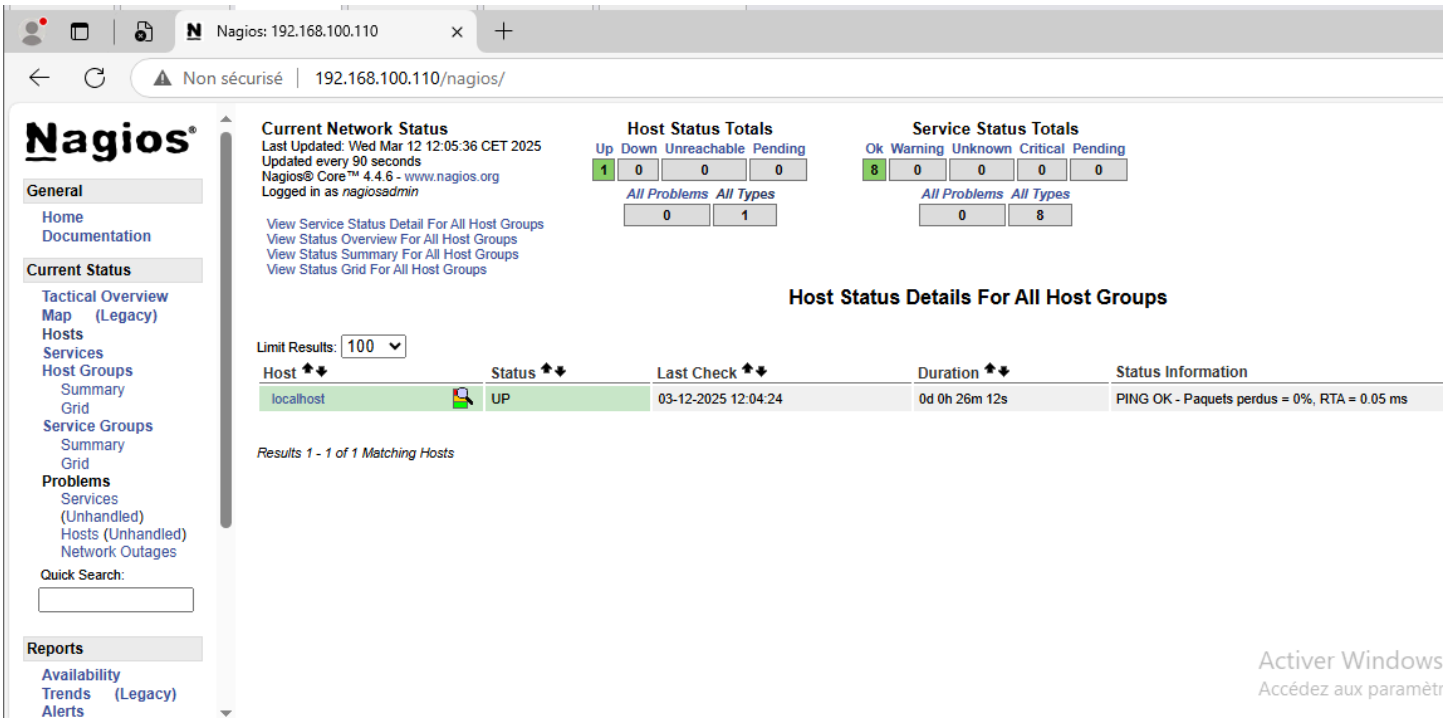


la machine Windows server Active Directory ou un Windows 10

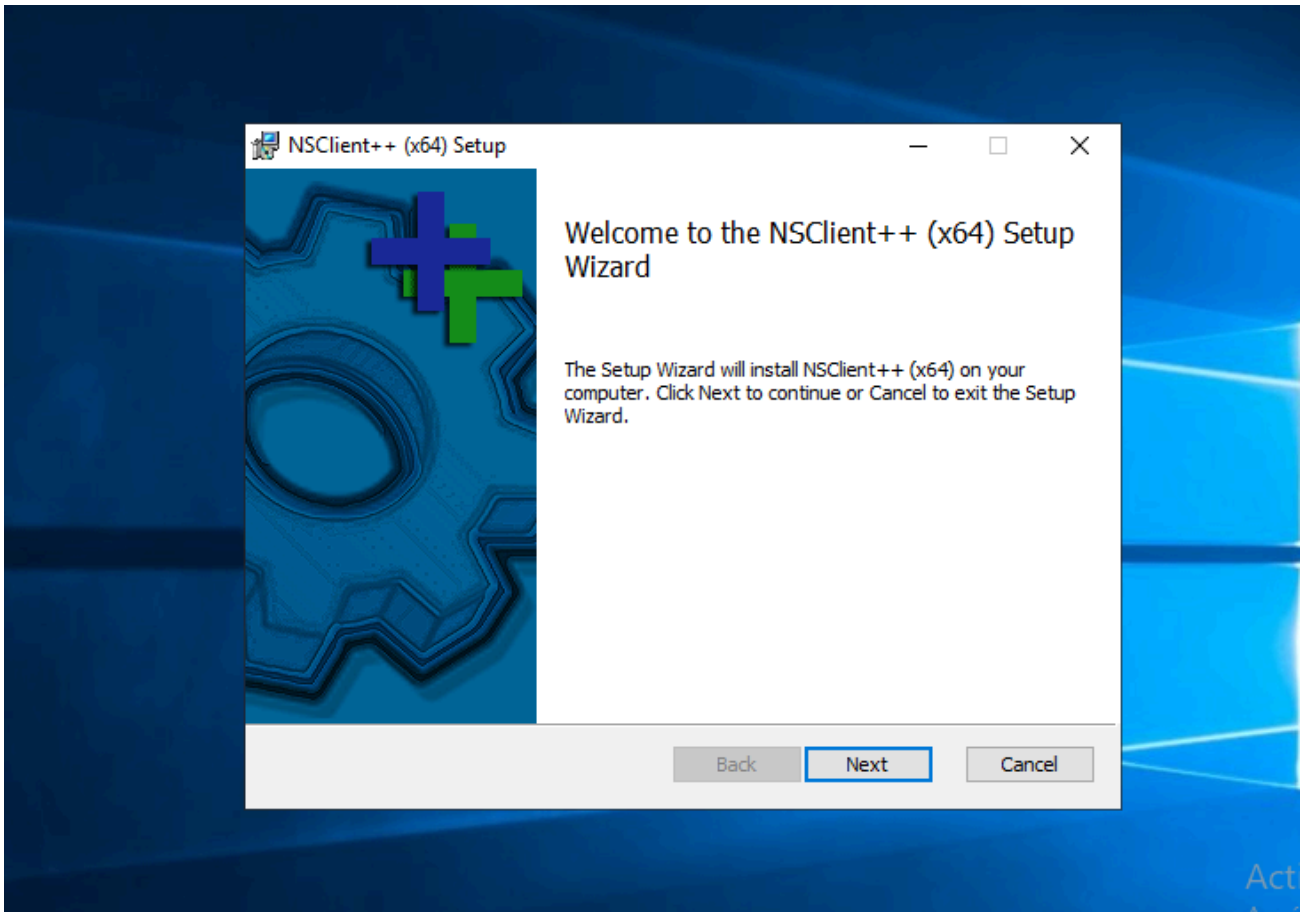
1. Lancer le navigateur et tapez l'ip de nagios **192.168.100.110/nagios**
2. **mettre le mot de passe que vous avez créer**

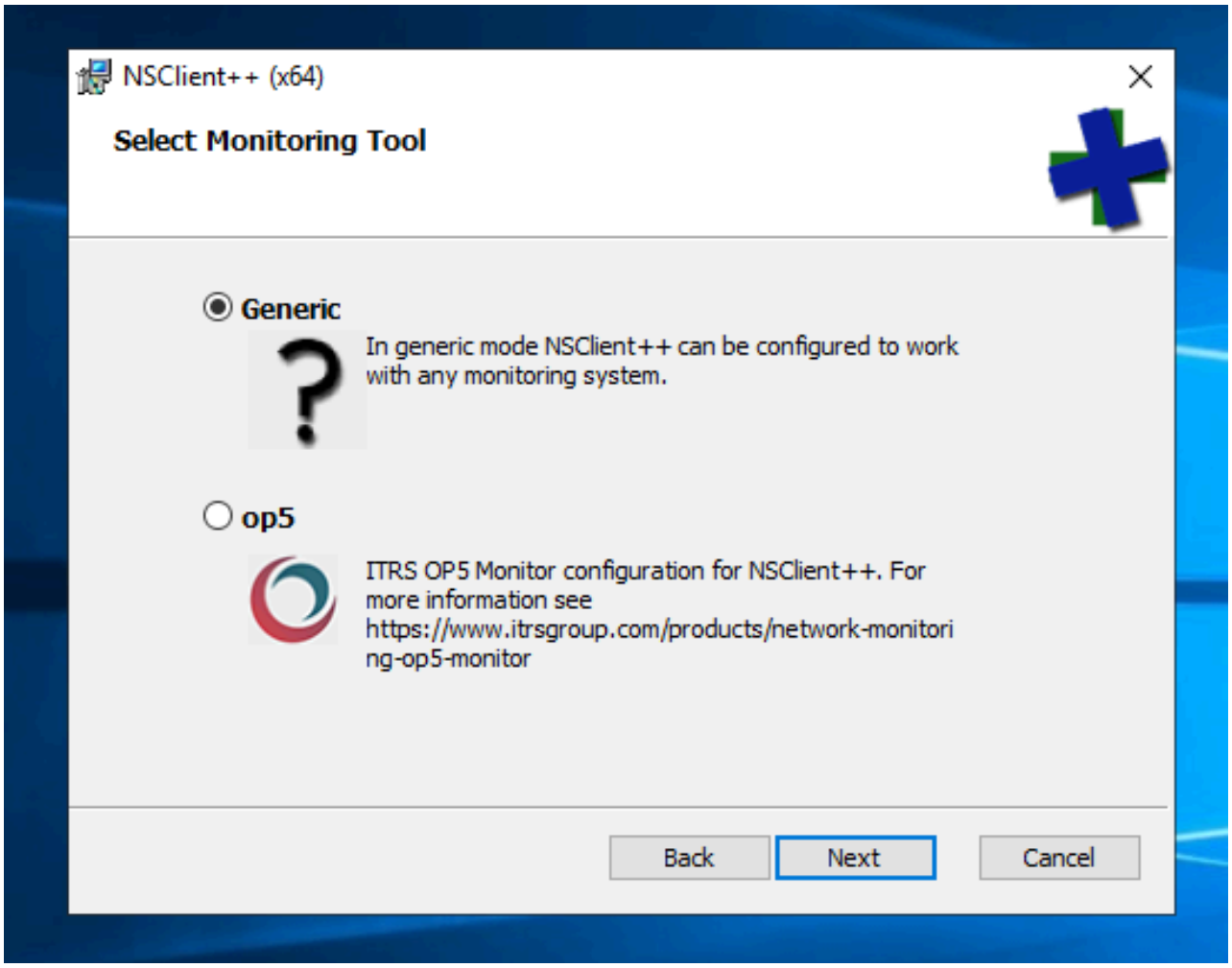


1. Faire un clic sur Hosts

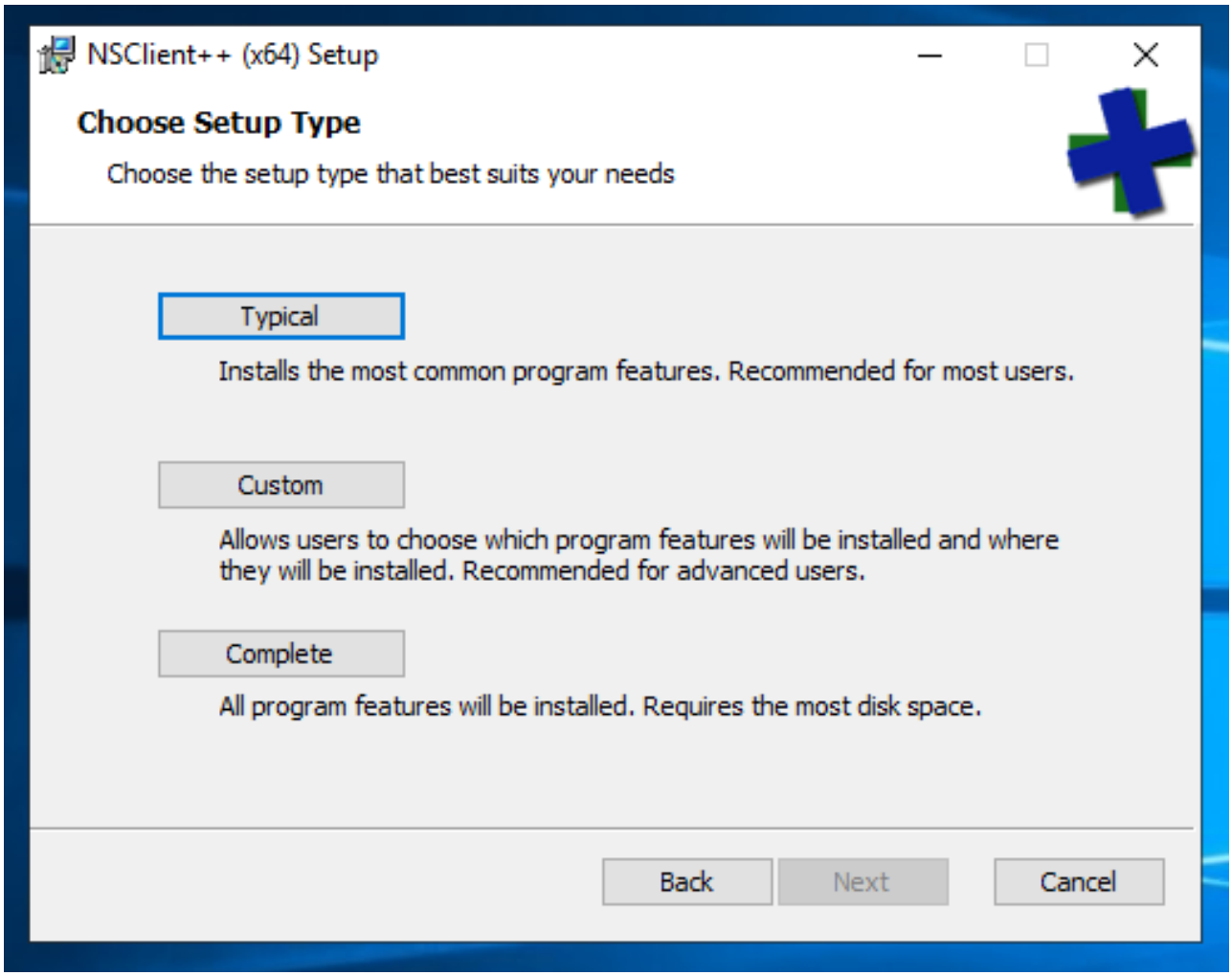


1. **Installer NSClient++ sur le serveur Windows**
2. Téléchargez NSClient++ depuis [Sortie 0.6.9 · mickem/nscp · Lien avec GitHub]
3. Suivre après....

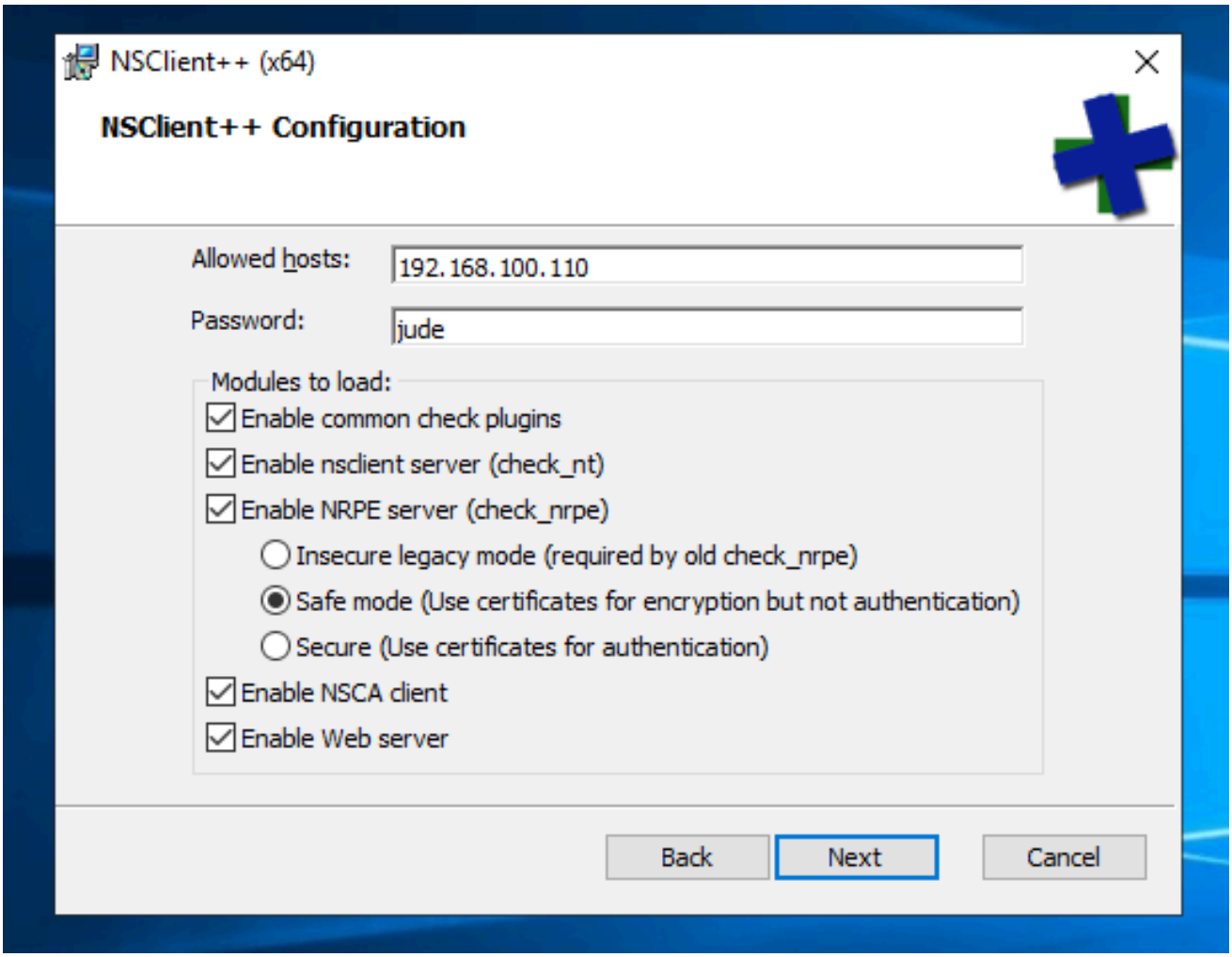


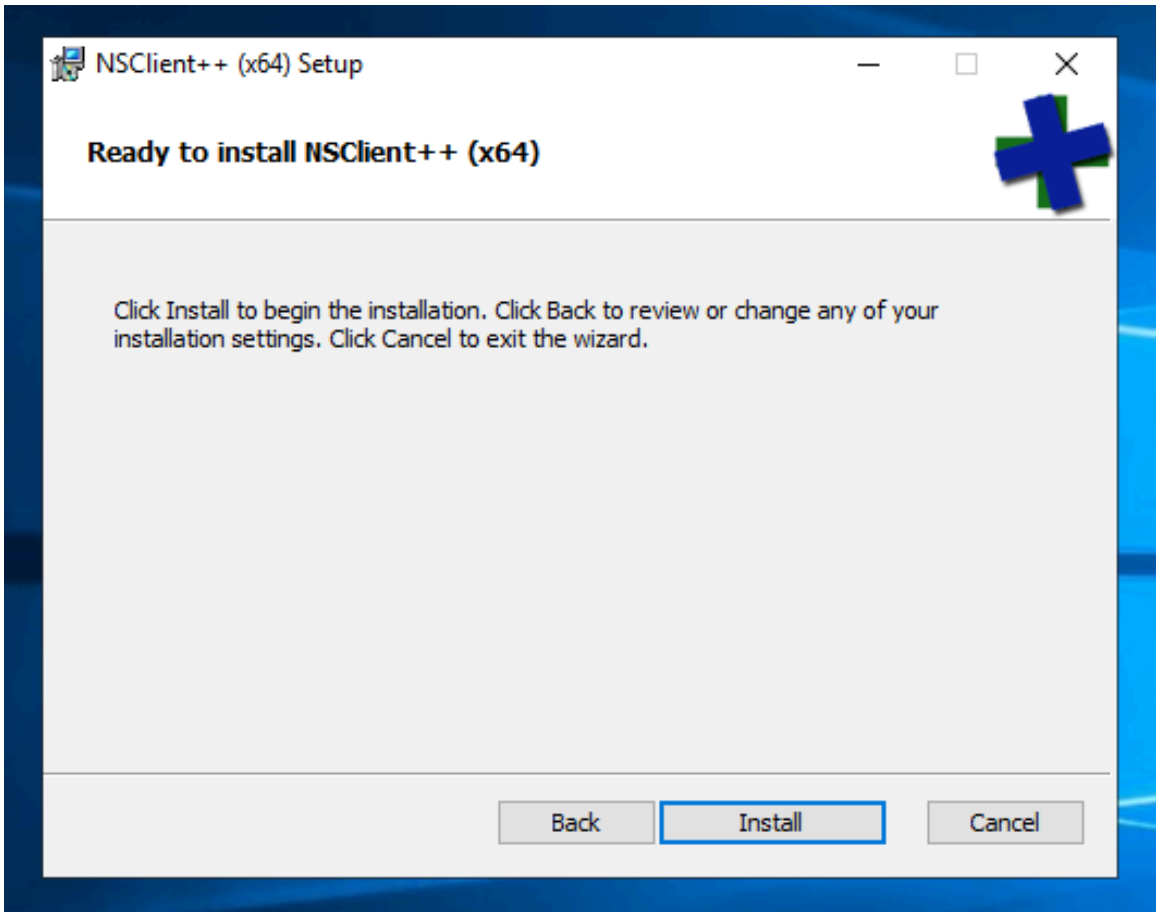


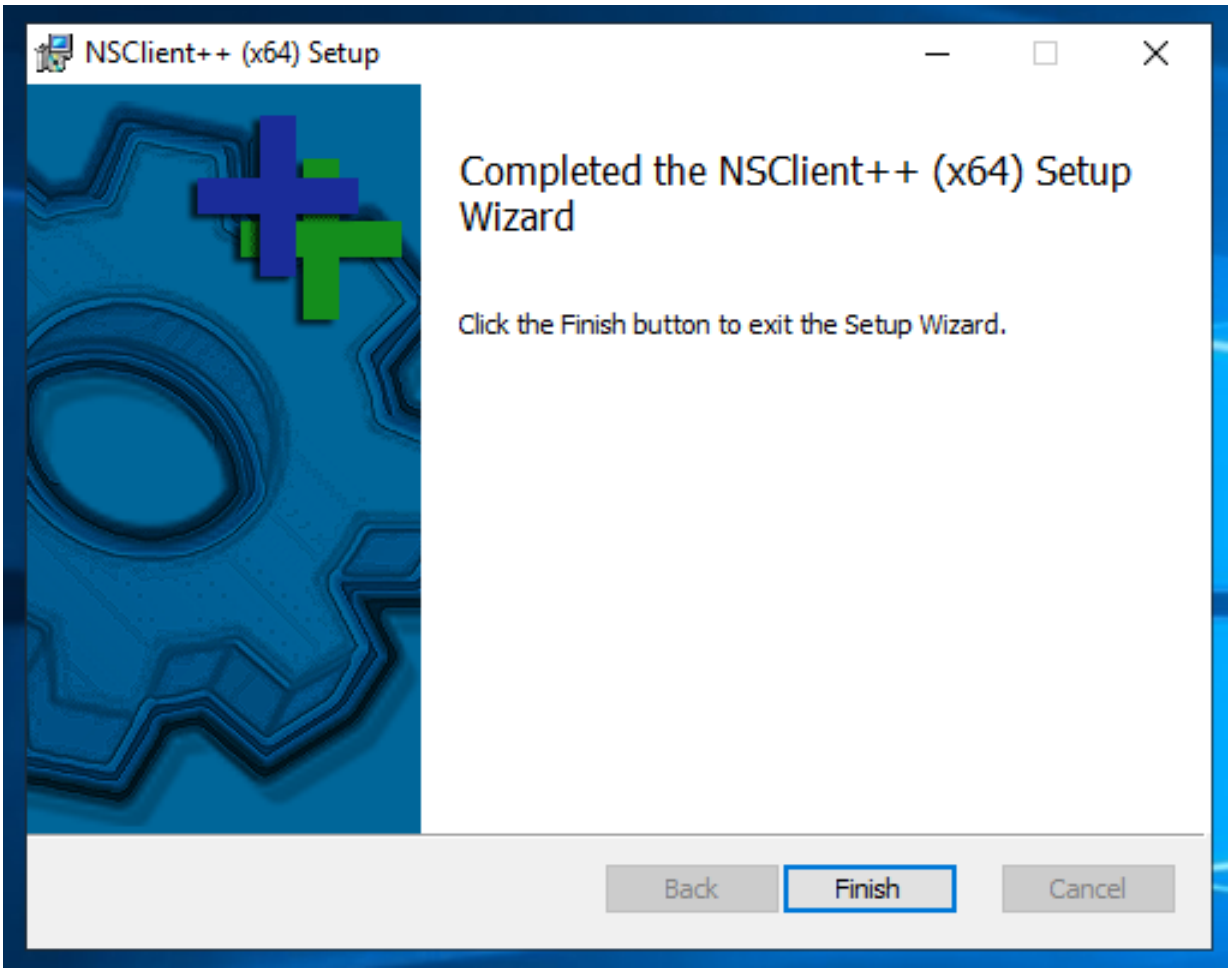
1. Choisir complète



1. Mettre l'adresse IP du serveur Nagios, le mot de passe n'est pas obligatoire


NSClient++ (x64)
NSClient++ Configuration
Allowed hosts:
192.168.100.110
Password:
jude
Modules to load:
Enable common check plugins
Enable nsclient server (check_nt)
Enable NRPE server (check_nrpe)
Insecure legacy mode (required by old check_nrpe)
Safe mode (Use certificates for encryption but not authentication)
Secure (Use certificates for authentication)
Enable NSCA client
Enable Web server
Back
Next
Cancel


NSClient++ (x64) Setup
Ready to install NSClient++ (x64)
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.
Back
Install
Cancel



1. Aller dans **l'exploirateur de fichier** / **disque dur C**/ **Programme** / **NSclient+++** puis dans le fichier **nsclient.ini**

; in flight - TODO

[/modules]

; Undocumented key

NSClientServer = enabled

; Undocumented key

CheckEventLog = disabled

; Undocumented key

CheckNSCP = disabled

; Undocumented key

CheckSystem = 1

; Undocumented key

CheckDisk = 1

; Undocumented key

NSCAClient = enabled

; Undocumented key

WEBServer = enabled

; Undocumented key

NRPEServer = enabled

; Undocumented key

CheckExternalScripts = disabled

; Undocumented key

CheckHelpers = disabled

; Undocumented key

NRPEListener = 1

; in flight - TODO

[/settings/NRPE/server]

; Undocumented key

verify mode = peer-cert

; Undocumented key

ssl options = no-sslv2,no-sslv3

; Undocumented key

insecure = false

; in flight - TODO

[/settings/default]

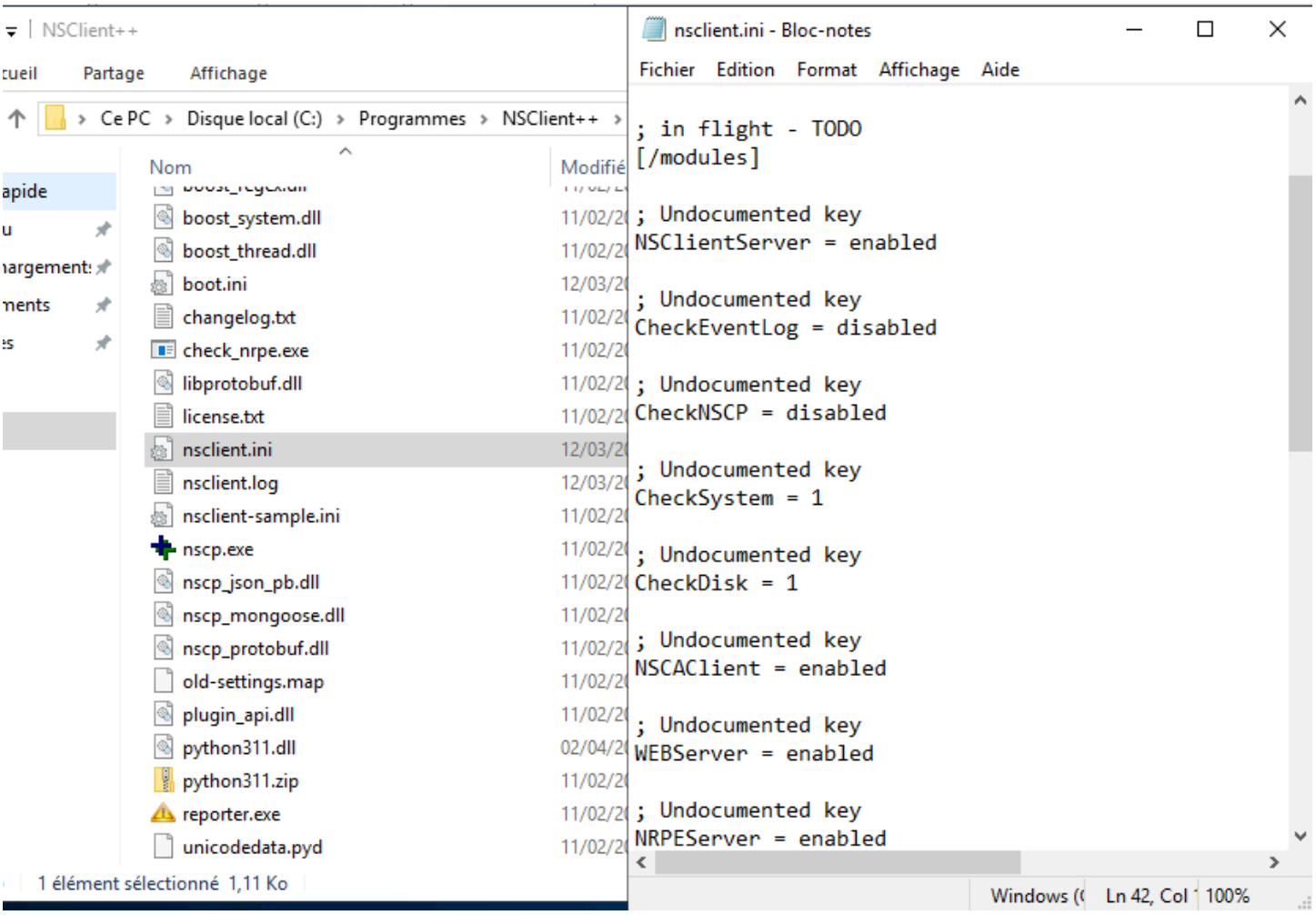
Enable

; Undocumented key

password = jude

; Undocumented key

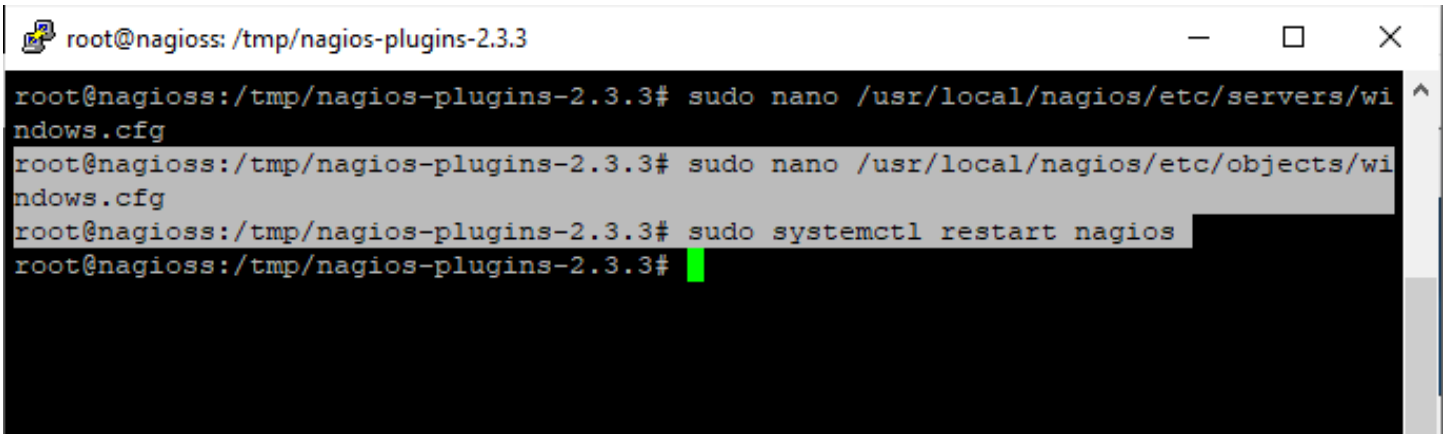
allowed hosts = 192.168.100.110

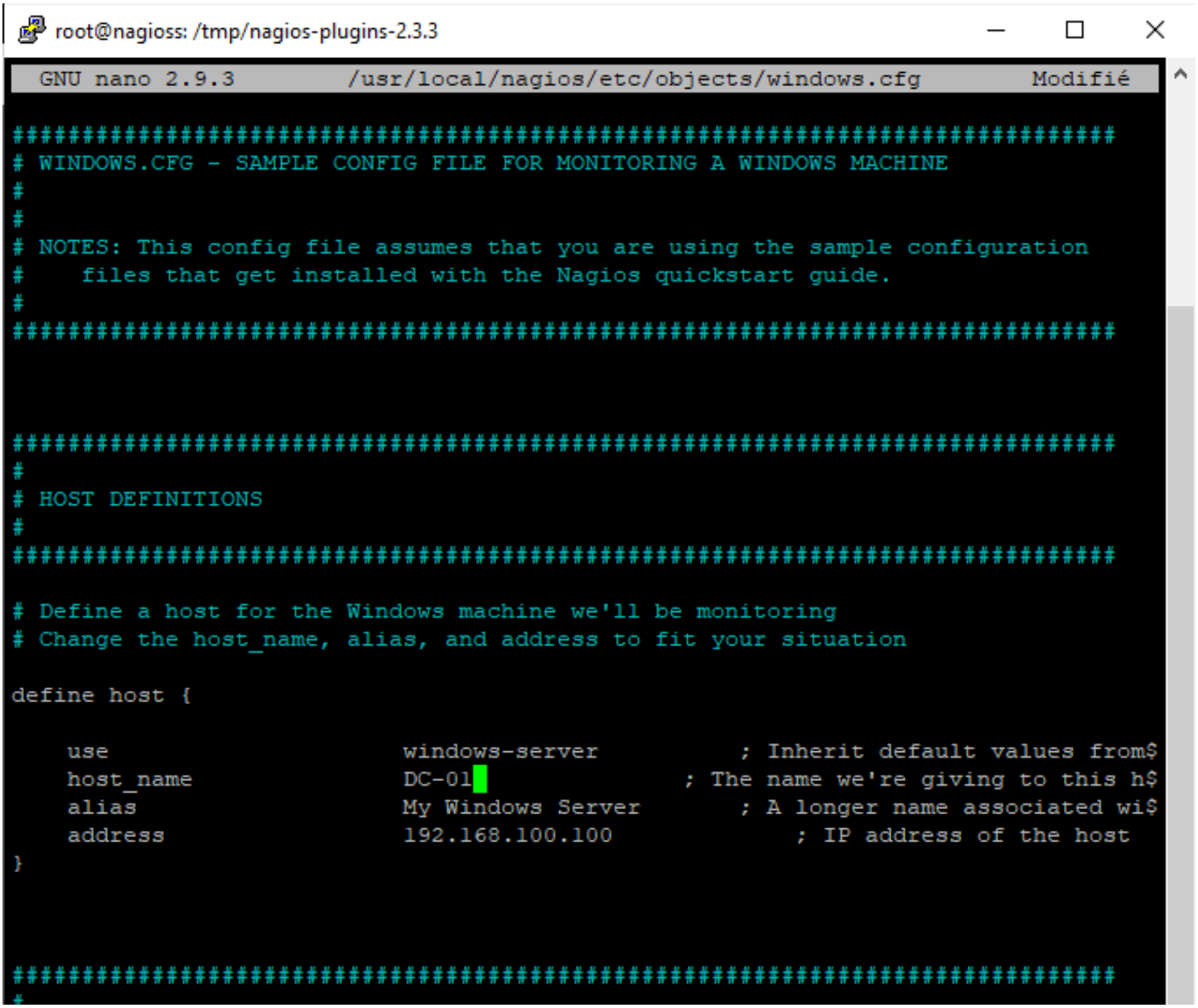


la machine nagios1

1. Ajouter l'hôte Windows dans Nagios

- **sudo nano /usr/local/nagios/etc/objets/windows.cfg :**

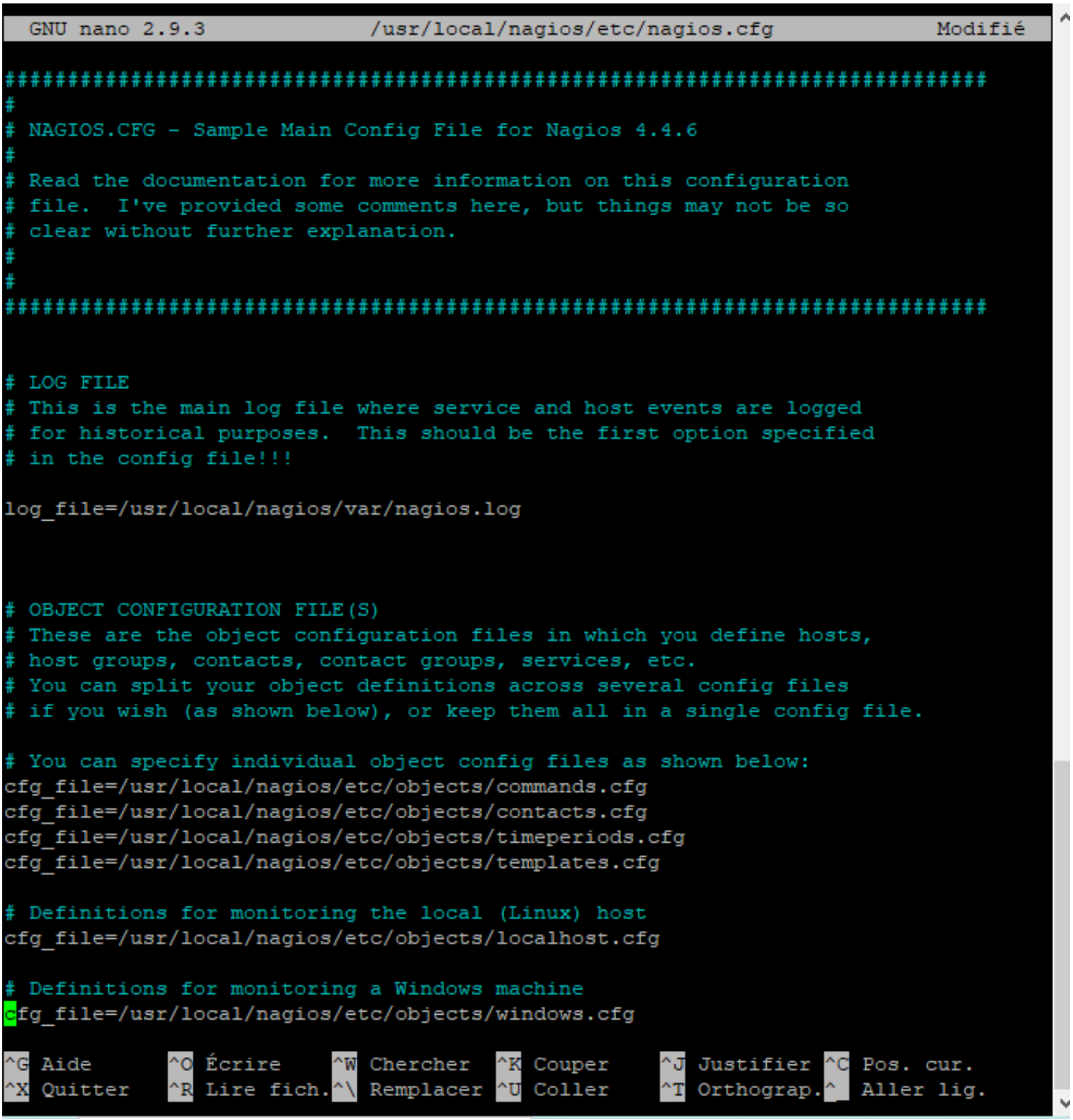



root@nagioss: /tmp/nagios-plugins-2.3.3
GNU nano 2.9.3 /usr/local/nagios/etc/objects/windows.cfg Modifié
# WINDOWS.CFG - SAMPLE CONFIG FILE FOR MONITORING A WINDOWS MACHINE
# NOTES: This config file assumes that you are using the sample configuration
# files that get installed with the Nagios quickstart guide.
# HOST DEFINITIONS
# Define a host for the Windows machine we'll be monitoring
# Change the host_name, alias, and address to fit your situation
define host {
use windows-server ; Inherit default values from$
host_name DC-01 ; The name we're giving to this h$
alias My Windows Server ; A longer name associated wi$
address 192.168.100.100 ; IP address of the host
}

```
#
# SERVICE DEFINITIONS
#
###############################################################################

# 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-01
    service_description     NSClient++ Version
    check_command           check_nt!CLIENTVERSION
}



# 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 {


^G Aide       ^O Écrire     ^W Chercher   ^K Couper     ^J Justifier  ^C Pos. cur.
^X Quitter    ^R Lire fich. ^\ Remplacer  ^U Coller     ^T Orthograp. ^_ Aller lig.
```

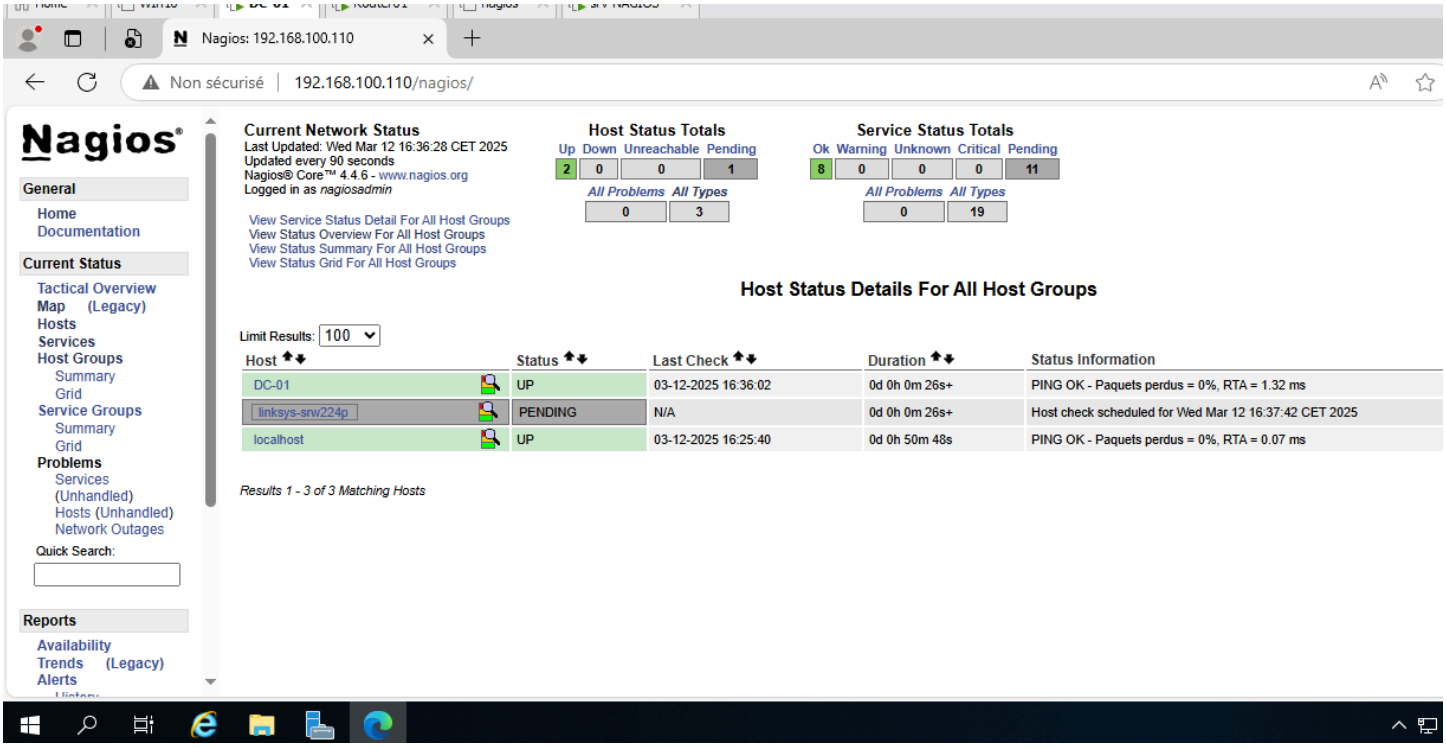
```
 GNU nano 2.9.3                          windows.cfg

# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               DC-01
    service_description     CPU Load
    check_command           check_nt!CPULOAD!-l 5,80,90
}



# 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-01
    service_description     Memory Usage
    check_command           check_nt!MEMUSE!-w 80 -c 90
}



# 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-01
    service_description     C:\ Drive Space
    check_command           check_nt!USEDDISKSPACE!-l c -w 80 -c 90
}



# Create a service for monitoring the W3SVC service

^G Aide       ^O Écrire     ^W Chercher   ^K Couper     ^J Justifier  ^C Pos. cur.  M-U Annuler   M-A Marquer
```

1. Aller dans le fichier nagios.cfg
- **sudo nano /usr/local/nagios/etc/nagios.cfg**
2. décommenter le ligne : **cfg_file=/usr/local/nagios/etc/objects/windows.cfg**



- **sudo systemctl restart nagios**
- **sudo systemctl status nagios**

1. **Accès à l'interface Web**

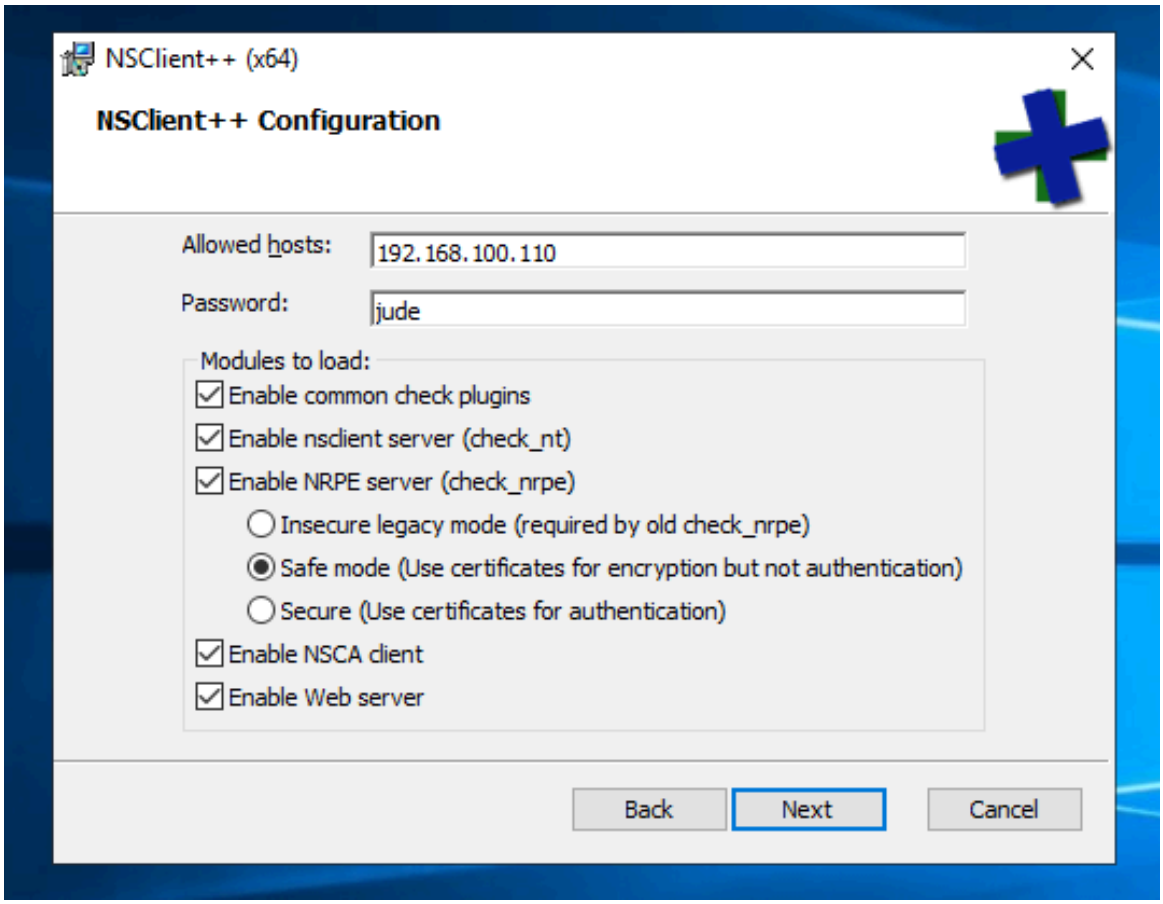


Vous avez maintenant un système Nagios fonctionnel sur Ubuntu 18.04. La prochaine étape consiste à affiner la configuration pour correspondre à vos besoins spécifiques en matière de supervision

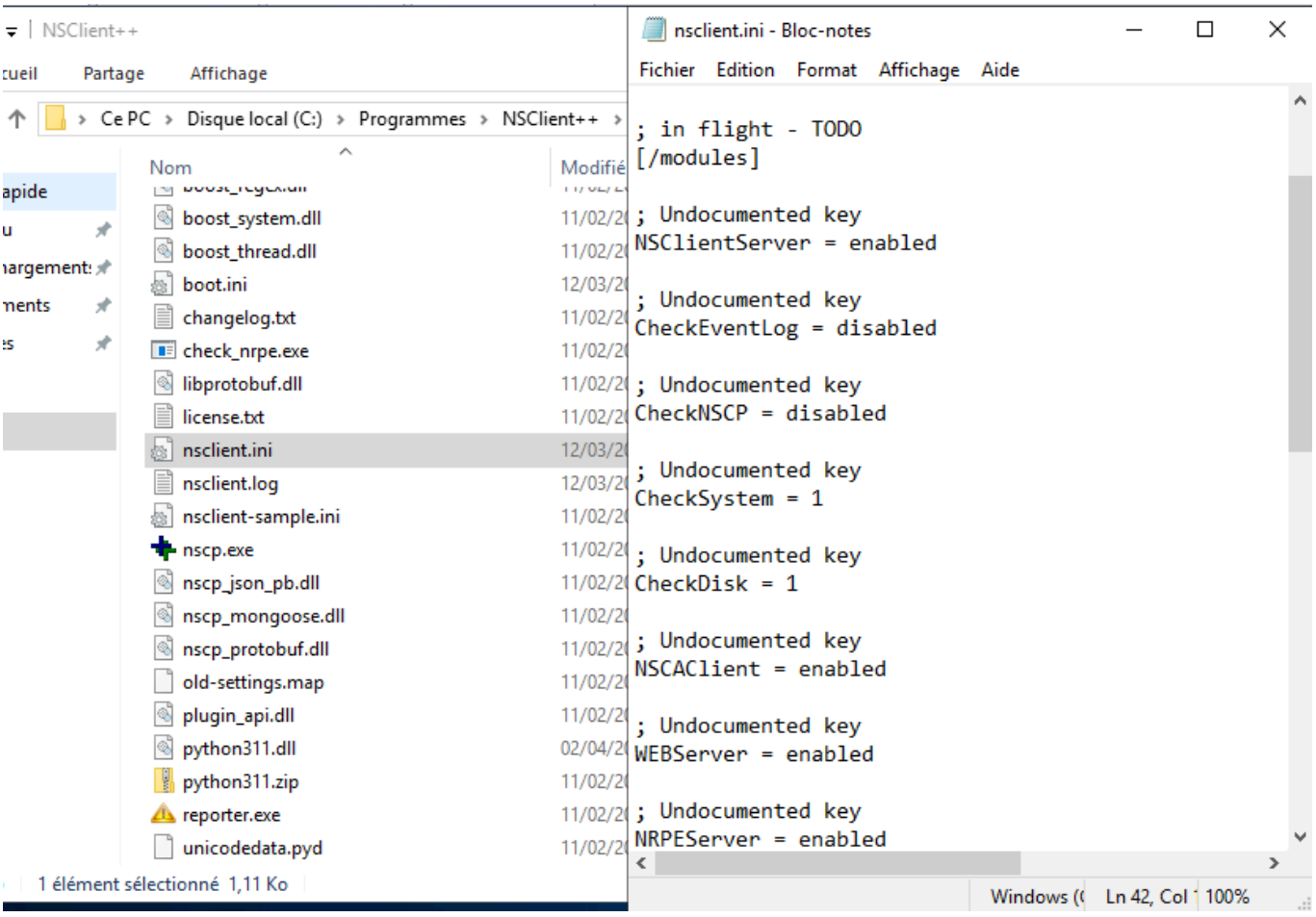
2ème Partie

Lancer la machine Windows server qui joue le rôle de routeur

1. **Installer le NSclient++ sur le routeur**
2. Choisir complète
3. Mettre l'adresse IP du serveur Nagios



1. Aller dans **l'exploirateur de fichier** / **disque dur C**/ **Programme** / **NSclient+++** puis dans le fichier **nsclient.ini**



<u>Lancer la machine nagios1 pour ajouter une autres machine</u>

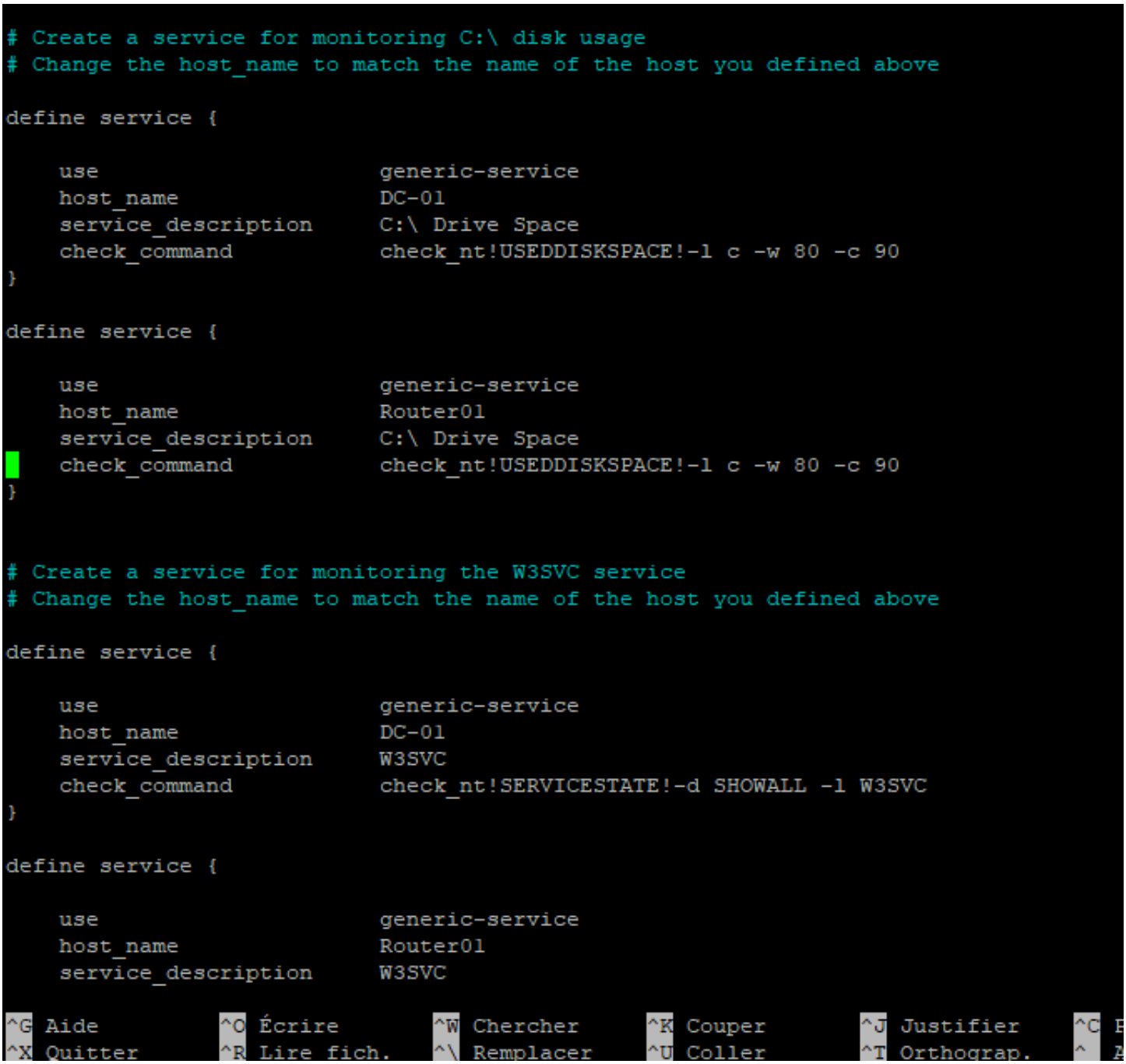
- **sudo nano /usr/local/nagios/etc/objects/windows.cfg**

root@nagios1: /home/admin1

```
root@nagios1:/home/admin1# nano /usr/local/nagios/etc/objects/windows.cfg
root@nagios1:/home/admin1#
```

```
  GNU nano 2.9.3                                        /usr/local/nagios/etc/objects/windows.cfg

##############################################################################
# WINDOWS.CFG - SAMPLE CONFIG FILE FOR MONITORING A WINDOWS MACHINE
#
#
# NOTES: This config file assumes that you are using the sample configuration
#    files that get installed with the Nagios quickstart guide.
#
##############################################################################



##############################################################################
#
# HOST DEFINITIONS
#
##############################################################################

# Define a host for the Windows machine we'll be monitoring
# 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-01               ; The name we're giving to this host
    alias                   My Windows Server       ; A longer name associated with the host
    address                 192.168.100.100             ; IP address of the host
}

define host {

    use                     windows-server          ; Inherit default values from $
    host_name               Router01              ; The name we're giving to this ho$
    alias                   My router       ; A longer name associated wit$
    address                 192.168.100.254             ; IP address of the host
}
```

```
  GNU nano 2.9.3                                                  /usr/local/nagio


# 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-01
    service_description     NSClient++ Version
    check_command           check_nt!CLIENTVERSION
}

define service {

    use                     generic-service
    host_name               Router01
    service_description     NSClient++ Version
    check_command           check_nt!CLIENTVERSION
}


# 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-01
    service_description     Uptime
    check_command           check_nt!UPTIME
}

define service {

    use                     generic-service
    host_name               Router01
    service_description     Uptime
    check_command           check_nt!UPTIME
```



```
# 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-01
    service_description     C:\ Drive Space
    check_command           check_nt!USEDDISKSPACE!-l c -w 80 -c 90
}

define service {

    use                     generic-service
    host_name               Router01
    service_description     C:\ Drive Space
    check_command           check_nt!USEDDISKSPACE!-l c -w 80 -c 90
}


# 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-01
    service_description     W3SVC
    check_command           check_nt!SERVICESTATE!-d SHOWALL -l W3SVC
}

define service {

    use                     generic-service
    host_name               Router01
    service_description     W3SVC

^G Aide         ^O Écrire       ^W Chercher     ^K Couper       ^J Justifier    ^C
^X Quitter      ^R Lire fich.   ^\ Remplacer    ^U Coller       ^T Orthograp.   ^
```

- **sudo systemctl restart nagios**
- **sudo systemctl status nagios**

```
root@nagios1:/home/admin1# sudo systemctl restart nagios
root@nagios1:/home/admin1# sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
     Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enab
     Active: active (running) since Thu 2025-03-13 13:11:33 CET; 9s ago
       Docs: https://www.nagios.org/documentation
    Process: 5003 ExecStopPost=/bin/rm -f /usr/local/nagios/var/rw/nagios.cmd (code=e
    Process: 4999 ExecStop=/bin/kill -s TERM ${MAINPID} (code=exited, status=0/SUCCES
    Process: 5065 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nag
    Process: 5064 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/
   Main PID: 5066 (nagios)
      Tasks: 6 (limit: 4620)
     CGroup: /system.slice/nagios.service
             ├─5066 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─5067 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na
             ├─5068 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na
             ├─5069 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na
             ├─5070 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na
             └─5071 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

mars 13 13:11:33 nagios1 systemd[1]: Started Nagios Core 4.4.6.
mars 13 13:11:33 nagios1 nagios[5066]: qh: core query handler registered
```