# Lesson 09 Demo 01

# Installing and Configuring Nagios Monitoring Tool

**Objective:** To install and configure Nagios monitoring tool for system health and network monitoring to ensure thorough oversight and management of infrastructure

**Tools required:** Ubuntu 22.04 instance

**Prerequisites:** Ensure Apache is installed

Steps to be followed:

1. Install and configure Nagios Core
2. Verify and enable Nagios
3. Upgrade to the latest version of Nagios

## Step 1: Install and configure Nagios Core

1.1 Install all the required packages using the following commands:
**sudo apt install wget unzip curl openssl build-essential libgd-dev libssl-dev libapache2-mod-php php-gd php apache2 -y**

```
labsuser@ip-172-31-40-202:~$ sudo apt install wget unzip curl openssl build-essential libgd-dev libssl-dev libapache2-mod-php php-gd php apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.9ubuntu3).
build-essential set to manually installed.
wget is already the newest version (1.21.2-2ubuntu1).
unzip is already the newest version (6.0-26ubuntu3.1).
You might want to run 'apt --fix-broken install' to correct these.
The following packages have unmet dependencies:
 apache2 : Depends: apache2-bin (= 2.4.52-1ubuntu4.7) but it is not going to be installed
           Depends: apache2-data (= 2.4.52-1ubuntu4.7) but it is not going to be installed
           Depends: apache2-utils (= 2.4.52-1ubuntu4.7) but it is not going to be installed
 curl : Depends: libcurl4 (= 7.81.0-1ubuntu1.15) but 7.81.0-1ubuntu1.13 is to be installed
 libapache2-mod-php : Depends: libapache2-mod-php8.1 but it is not going to be installed
 libgd-dev : Depends: libpng-dev but it is not going to be installed
             Depends: libz-dev
             Depends: libjpeg-dev
             Depends: libfreetype6-dev
             Depends: libxpm-dev but it is not going to be installed
             Depends: libx11-dev but it is not going to be installed
             Depends: libxt-dev but it is not going to be installed
             Depends: libfontconfig-dev but it is not going to be installed
             Depends: libvpx-dev but it is not going to be installed
             Depends: libtiff-dev but it is not going to be installed
 libssl-dev : Depends: libssl3 (= 3.0.2-0ubuntu1.14) but 3.0.2-0ubuntu1.10 is to be installed
 php : Depends: php8.1 but it is not going to be installed
 php-gd : Depends: php-common but it is not going to be installed
          Depends: php8.1-gd but it is not going to be installed
 shim-signed : Depends: grub-efi-amd64-signed (>= 1.187.2~) but it is not going to be installed or
                        grub-efi-arm64-signed (>= 1.187.2~) but it is not installable
```

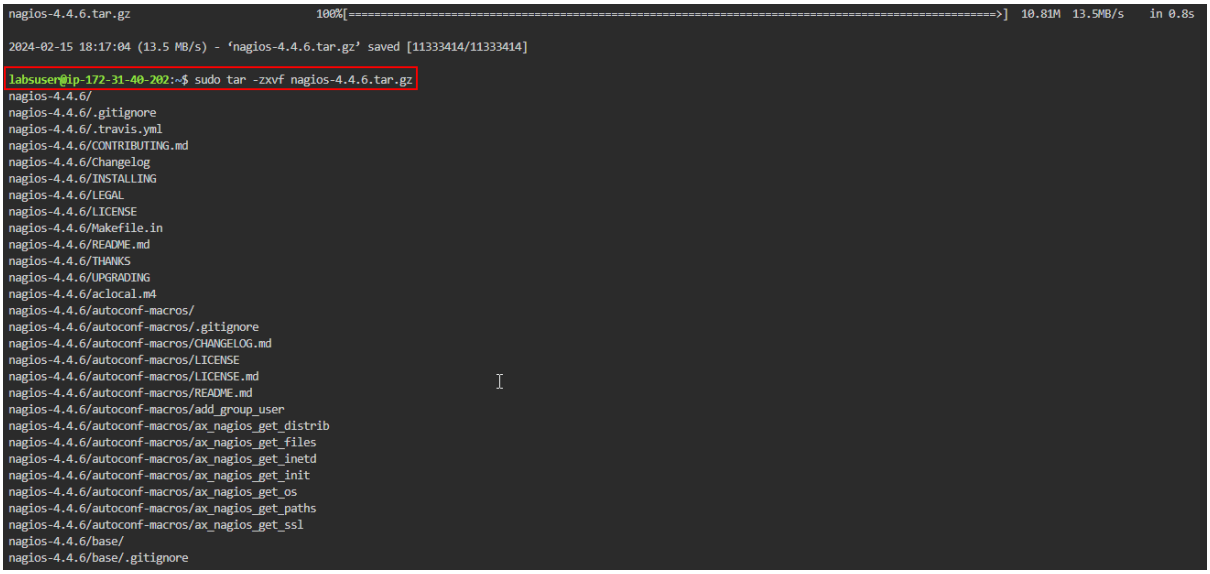
**sudo apt --fix-broken install**

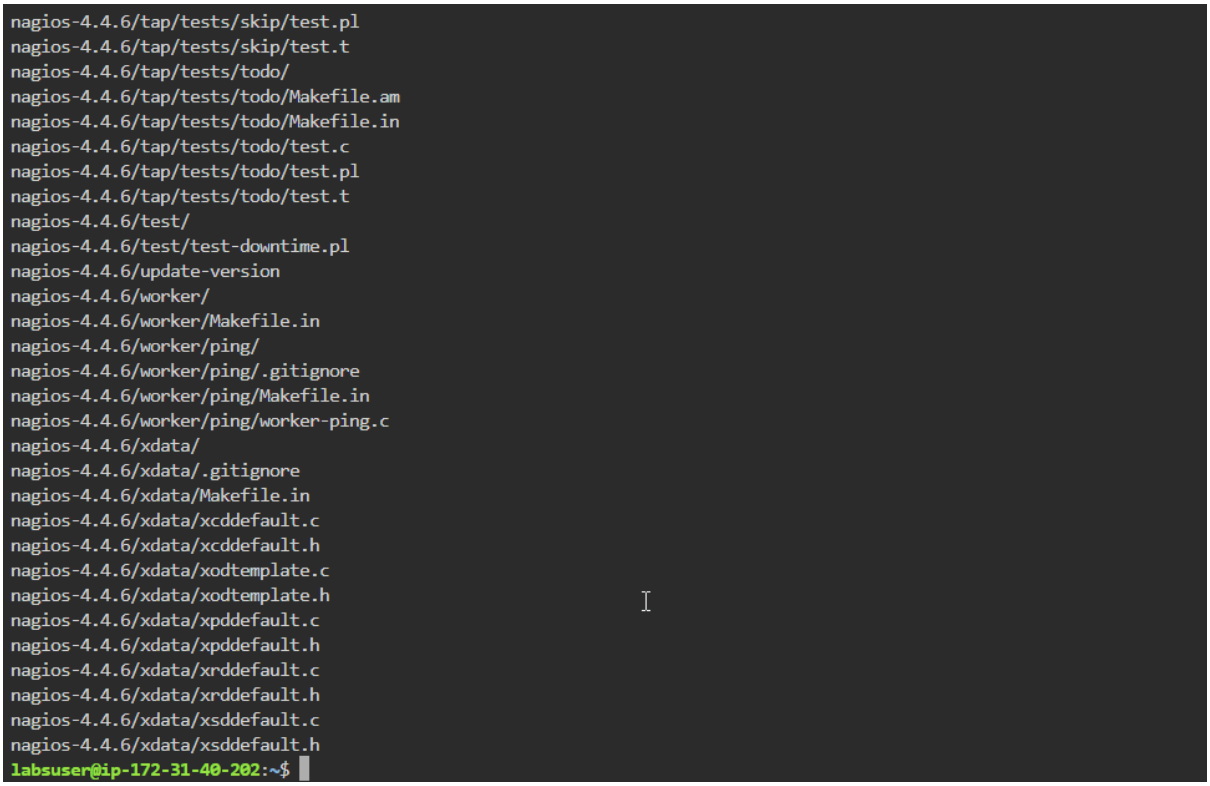
```
labsuser@ip-172-31-40-202:~$ sudo apt --fix-broken install
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Correcting dependencies... Done
The following additional packages will be installed:
  grub-efi-amd64-signed
The following NEW packages will be installed:
  grub-efi-amd64-signed
0 upgraded, 1 newly installed, 0 to remove and 208 not upgraded.
Need to get 0 B/1389 kB of archives.
After this operation, 7185 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Selecting previously unselected package grub-efi-amd64-signed.
(Reading database ... 218894 files and directories currently installed.)
Preparing to unpack .../grub-efi-amd64-signed_1.187.6+2.06-2ubuntu14.4_amd64.deb ...
Unpacking grub-efi-amd64-signed (1.187.6+2.06-2ubuntu14.4) ...
Setting up grub-efi-amd64-signed (1.187.6+2.06-2ubuntu14.4) ...
Installing grub to /boot/efi.
Installing for x86_64-efi platform.
grub-install: warning: EFI variables cannot be set on this system.
grub-install: warning: You will have to complete the GRUB setup manually.
Installation finished. No error reported.
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.
```

1.2 Download Nagios Core setup files using the following command:
**wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz**

```
labsuser@ip-172-31-40-202:~$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
--2024-02-15 18:17:03--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11333414 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'

nagios-4.4.6.tar.gz                 100%[==========================================================================================>]  10.81M  13.5MB/s    in 0.8s

2024-02-15 18:17:04 (13.5 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333414/11333414]

labsuser@ip-172-31-40-202:~$
```

1.3 Extract the downloaded files using the following command:
**sudo tar -zxvf nagios-4.4.6.tar.gz**



1.4 Navigate to the setup directory using the following command:
**cd nagios-4.4.6**

```
nagios-4.4.6/xdata/.gitignore
nagios-4.4.6/xdata/Makefile.in
nagios-4.4.6/xdata/xcddefault.c
nagios-4.4.6/xdata/xcddefault.h
nagios-4.4.6/xdata/xodtemplate.c
nagios-4.4.6/xdata/xodtemplate.h
nagios-4.4.6/xdata/xpddefault.c
nagios-4.4.6/xdata/xpddefault.h
nagios-4.4.6/xdata/xrddefault.c
nagios-4.4.6/xdata/xrddefault.h
nagios-4.4.6/xdata/xsddefault.c
nagios-4.4.6/xdata/xsddefault.h
labsuser@ip-172-31-40-202:~$ cd nagios-4.4.6
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.5 Run the Nagios Core configure script using the following command:
**sudo ./configure**

```
labsuser@ip-172-31-40-202:~$ cd nagios-4.4.6
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo ./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 C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes
checking for unistd.h... yes
checking arpa/inet.h usability... yes
checking arpa/inet.h presence... yes
```

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

 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:  /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.

labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.6 Compile the main program and CGIs using the following command:
**sudo make all**

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

labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo make all
cd ./base && make
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
nagios.c: In function 'main':
nagios.c:611:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  611 |                         asprintf(&mac->x[MACRO_PROCESSSTARTTIME], "%llu", (unsigned long long)program_start);
      |                         ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
nagios.c:841:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  841 |                         asprintf(&mac->x[MACRO_EVENTSTARTTIME], "%llu", (unsigned long long)event_start);
      |                         ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
nagios.c: In function 'nagios_core_worker':
nagios.c:176:17: warning: ignoring return value of 'read' declared with attribute 'warn_unused_result' [-Wunused-result]
  176 |                 read(sd, response + 3, sizeof(response) - 4);
      |                 ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
nagios.c: In function 'test_path_access':
nagios.c:122:17: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  122 |                 asprintf(&path, "%s/%s", p, program);
      |                 ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE   -c -o broker.o broker.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE   -c -o nebmods.o nebmods.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE   -c -o query-handler.o query-handler.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
workers.c: In function 'handle_worker_result':
workers.c:801:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  801 |                         asprintf(&error_reason, "timed out after %.2fs", tv_delta_f(&wpres.start, &wpres.stop));
      |                         ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
workers.c:804:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
  804 |                         asprintf(&error_reason, "died by signal %d%s after %.2f seconds",
```

```
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/html'
(cd angularjs && unzip -u angular-1.3.9.zip)
Archive:  angular-1.3.9.zip
(cd angularjs && unzip -u ui-utils-0.2.3.zip)
Archive:  ui-utils-0.2.3.zip
125cb72d692f9f116454f338f6a5471f3088f9b9
if [ ! -d bootstrap-3.3.7 ] ; then mkdir -p bootstrap-3.3.7 && ( cd bootstrap-3.3.7 && unzip -u ../bootstrap-3.3.7-dist.zip && mv dist/* . && rmdir dist ) ; fi
mkdir -p d3
(cd d3 && unzip -u ../d3-3.5.17.zip)
Archive:  ../d3-3.5.17.zip
make[1]: Leaving directory '/home/labsuser/nagios-4.4.6/html'
if [ xyes = xyes ]; then \
        cd ./module && make; \
fi
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/module'
make[1]: Nothing to be done for 'all'.
make[1]: Leaving directory '/home/labsuser/nagios-4.4.6/module'
cd ./worker && make all
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/worker'
cd ./ping && make all
make[2]: Entering directory '/home/labsuser/nagios-4.4.6/worker/ping'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/home/labsuser/nagios-4.4.6/worker/ping'
make[1]: Leaving directory '/home/labsuser/nagios-4.4.6/worker'

*** 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):
```

1.7 Create and install group and user using the following command:
**sudo make install-groups-users**

```
*** 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.

labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo make install-groups-users
Group nagios already exists
User nagios already exists
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.8 Add www-data directories user to the Nagios group using the following command:
**sudo usermod -a -G nagios www-data**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo make install-groups-users
Group nagios already exists
User nagios already exists
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo usermod -a -G nagios www-data
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.9 Use the following command to install Nagios:
**sudo make install**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo make install-groups-users
Group nagios already exists
User nagios already exists
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo usermod -a -G nagios www-data
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiostats /usr/local/nagios/bin
make[1]: Leaving directory '/home/labsuser/nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/labsuser/nagios-4.4.6/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
        /usr/bin/install -c -s -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
make[2]: Leaving directory '/home/labsuser/nagios-4.4.6/cgi'
make[1]: Leaving directory '/home/labsuser/nagios-4.4.6/cgi'
cd ./html && make install
make[1]: Entering directory '/home/labsuser/nagios-4.4.6/html'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/media
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/stylesheets
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/contexthelp
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs/images
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/js
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images
```

```
make install-exfoliation
make[1]: Entering directory '/home/labsuser/nagios-4.4.6'

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

make[1]: Leaving directory '/home/labsuser/nagios-4.4.6'
make install-basic
make[1]: Entering directory '/home/labsuser/nagios-4.4.6'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/archives
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/spool/checkresults
chmod g+s /usr/local/nagios/var/spool/checkresults

*** 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 '/home/labsuser/nagios-4.4.6'
```

1.10 Initialize all the installation configuration scripts using the following command:
**sudo make install-init**

```
*** 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 '/home/labsuser/nagios-4.4.6'
labsuser@ip-172-31-40-202:~/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/system/nagios.service
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.11 Install and configure permissions on the config's directory using the following command:

**sudo make install-commandmode**

```
labsuser@ip-172-31-40-202:~/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/system/nagios.service
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo 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 ***

labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.12 Install sample config files using the following command:

**sudo make install-config**

```
labsuser@ip-172-31-40-202:~/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/system/nagios.service
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo 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 ***

labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo 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.

labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.13 Enable Apache rewrite mode using the following command:

**sudo a2enmod rewrite**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

**Note**: Ensure you have Apache installed before executing the above command

1.14 Enable CGI config using the following command:

**sudo a2enmod cgi**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo a2enmod cgi
Your MPM seems to be threaded. Selecting cgid instead of cgi.
Enabling module cgid.
To activate the new configuration, you need to run:
  systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.15 Restart the Apache service using the following command:

**sudo systemctl restart apache2**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo a2enmod cgi
Your MPM seems to be threaded. Selecting cgid instead of cgi.
Enabling module cgid.
To activate the new configuration, you need to run:
  systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

1.16 Create a user and set the password when prompted using the following command:

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

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo a2enmod cgi
Your MPM seems to be threaded. Selecting cgid instead of cgi.
Enabling module cgid.
To activate the new configuration, you need to run:
  systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users admin
New password:
Re-type new password:
Adding password for user admin
labsuser@ip-172-31-40-202:~/nagios-4.4.6$
```

## Step 2: Verify and enable Nagios

2.1 Download the Nagios Core plugin using the following command:

**cd ~/**

**wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo a2enmod cgi
Your MPM seems to be threaded. Selecting cgid instead of cgi.
Enabling module cgid.
To activate the new configuration, you need to run:
  systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo systemctl restart apache2
labsuser@ip-172-31-40-202:~/nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users admin
New password:
Re-type new password:
Adding password for user admin
labsuser@ip-172-31-40-202:~/nagios-4.4.6$  cd ~/
 wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
--2024-02-15 19:20:18--  https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: 'nagios-plugins-2.3.3.tar.gz'

nagios-plugins-2.3.3.tar.gz              100%[=============================================================================>]   2.65M  5.02MB/s    in 0.5s

2024-02-15 19:20:18 (5.02 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]

labsuser@ip-172-31-40-202:~$
```

2.2 Extract the downloaded plugin using the following command:

**sudo tar -zxvf nagios-plugins-2.3.3.tar.gz**

```
labsuser@ip-172-31-40-202:~/nagios-4.4.6$  cd ~/
 wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
--2024-02-15 19:20:18--  https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2782610 (2.7M) [application/x-gzip]
Saving to: 'nagios-plugins-2.3.3.tar.gz'

nagios-plugins-2.3.3.tar.gz              100%[=============================================================================>]   2.65M  5.02MB/s    in 0.5s

2024-02-15 19:20:18 (5.02 MB/s) - 'nagios-plugins-2.3.3.tar.gz' saved [2782610/2782610]

labsuser@ip-172-31-40-202:~$ sudo tar -zxvf nagios-plugins-2.3.3.tar.gz
nagios-plugins-2.3.3/
nagios-plugins-2.3.3/perlmods/
nagios-plugins-2.3.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.3.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.3.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.in
nagios-plugins-2.3.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile.am
nagios-plugins-2.3.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.3.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.3.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.3.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.3.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.3.3/perlmods/Makefile
nagios-plugins-2.3.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.3.3/perlmods/install_order
```

```
nagios-plugins-2.3.3/plugins-scripts/check_breeze.pl
nagios-plugins-2.3.3/plugins-scripts/check_log.sh
nagios-plugins-2.3.3/plugins-scripts/check_ssl_validity.pl
nagios-plugins-2.3.3/plugins-scripts/check_flexlm.pl
nagios-plugins-2.3.3/plugins-scripts/check_rpc.pl
nagios-plugins-2.3.3/plugins-scripts/check_oracle.sh
nagios-plugins-2.3.3/plugins-scripts/utils.pm.in
nagios-plugins-2.3.3/plugins-scripts/check_disk_smb.pl
nagios-plugins-2.3.3/plugins-scripts/t/
nagios-plugins-2.3.3/plugins-scripts/t/check_ifoperstatus.t
nagios-plugins-2.3.3/plugins-scripts/t/check_rpc.t
nagios-plugins-2.3.3/plugins-scripts/t/check_file_age.t
nagios-plugins-2.3.3/plugins-scripts/t/check_disk_smb.t
nagios-plugins-2.3.3/plugins-scripts/t/check_ifstatus.t
nagios-plugins-2.3.3/plugins-scripts/t/utils.t
nagios-plugins-2.3.3/plugins-scripts/check_mailq.pl
nagios-plugins-2.3.3/plugins-scripts/check_wave.pl
nagios-plugins-2.3.3/plugins-scripts/check_ircd.pl
nagios-plugins-2.3.3/plugins-scripts/utils.sh.in
nagios-plugins-2.3.3/plugins-scripts/check_ifstatus.pl
nagios-plugins-2.3.3/plugins-scripts/check_sensors.sh
nagios-plugins-2.3.3/pkg/
nagios-plugins-2.3.3/pkg/fedora/
nagios-plugins-2.3.3/pkg/fedora/requires
nagios-plugins-2.3.3/pkg/solaris/
nagios-plugins-2.3.3/pkg/solaris/preinstall
nagios-plugins-2.3.3/pkg/solaris/solpkg
nagios-plugins-2.3.3/pkg/solaris/pkginfo.in
nagios-plugins-2.3.3/pkg/solaris/pkginfo
nagios-plugins-2.3.3/pkg/redhat/
nagios-plugins-2.3.3/pkg/redhat/requires
```

2.3 Navigate to the plugin's directory using the following command:
**cd nagios-plugins-2.3.3/**

```
nagios-plugins-2.3.3/plugins-scripts/t/
nagios-plugins-2.3.3/plugins-scripts/t/check_ifoperstatus.t
nagios-plugins-2.3.3/plugins-scripts/t/check_rpc.t
nagios-plugins-2.3.3/plugins-scripts/t/check_file_age.t
nagios-plugins-2.3.3/plugins-scripts/t/check_disk_smb.t
nagios-plugins-2.3.3/plugins-scripts/t/check_ifstatus.t
nagios-plugins-2.3.3/plugins-scripts/t/utils.t
nagios-plugins-2.3.3/plugins-scripts/check_mailq.pl
nagios-plugins-2.3.3/plugins-scripts/check_wave.pl
nagios-plugins-2.3.3/plugins-scripts/check_ircd.pl
nagios-plugins-2.3.3/plugins-scripts/utils.sh.in
nagios-plugins-2.3.3/plugins-scripts/check_ifstatus.pl
nagios-plugins-2.3.3/plugins-scripts/check_sensors.sh
nagios-plugins-2.3.3/pkg/
nagios-plugins-2.3.3/pkg/fedora/
nagios-plugins-2.3.3/pkg/fedora/requires
nagios-plugins-2.3.3/pkg/solaris/
nagios-plugins-2.3.3/pkg/solaris/preinstall
nagios-plugins-2.3.3/pkg/solaris/solpkg
nagios-plugins-2.3.3/pkg/solaris/pkginfo.in
nagios-plugins-2.3.3/pkg/solaris/pkginfo
nagios-plugins-2.3.3/pkg/redhat/
nagios-plugins-2.3.3/pkg/redhat/requires
labsuser@ip-172-31-40-202:~$ cd nagios-plugins-2.3.3/
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$
```

2.4 Run the plugin configure script using the following command:
**sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios**

```
nagios-plugins-2.3.3/pkg/solaris/pkginfo
nagios-plugins-2.3.3/pkg/redhat/
nagios-plugins-2.3.3/pkg/redhat/requires
labsuser@ip-172-31-40-202:~$ cd nagios-plugins-2.3.3/
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether to disable maintainer-specific portions of Makefiles... yes
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for style of include used by make... GNU
checking dependency style of gcc... gcc3
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for Minix Amsterdam compiler... no
checking for ar... ar
checking for ranlib... ranlib
checking whether gcc and cc understand -c and -o together... yes
```

2.5 Compile Nagios Core plugins using the following command:
**sudo make**

```
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ sudo make
make  all-recursive
make[1]: Entering directory '/home/labsuser/nagios-plugins-2.3.3'
Making all in gl
make[2]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/gl'
make  all-recursive
make[3]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/gl'
make[4]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/gl'
make[4]: Nothing to be done for 'all-am'.
make[4]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/gl'
make[3]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/gl'
make[2]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/gl'
Making all in tap
make[2]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/tap'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/tap'
Making all in lib
make[2]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/lib'
Making all in .
make[3]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/lib'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/lib'
Making all in tests
make[3]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/lib/tests'
make[3]: Nothing to be done for 'all'.
make[3]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/lib/tests'
make[2]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/lib'
Making all in plugins
make[2]: Entering directory '/home/labsuser/nagios-plugins-2.3.3/plugins'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3/plugins'
```

**Note:** You may also install the above compiled plugins using **sudo make install** command.

2.6 Verify Nagios configuration using the following command:
**sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg**

```
make[1]: Leaving directory '/home/labsuser/nagios-plugins-2.3.3'
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$  sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
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.
```

2.7 Start the Nagios service using the following command:
**sudo systemctl start nagios**

```
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
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ sudo systemctl start nagios
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$
```

2.8 Enable Nagios service to run at system startup using the following command:
**sudo systemctl enable nagios**

```
Things look okay - No serious problems were detected during the pre-flight check
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ sudo systemctl start nagios
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ sudo systemctl enable nagios
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /lib/systemd/system/nagios.service.
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$
```

## Step 3: Upgrade to the latest version of Nagios

3.1 Navigate to the root directory using the following command:
**cd ..**

```
labsuser@ip-172-31-40-202:~/nagios-plugins-2.3.3$ cd ..
labsuser@ip-172-31-40-202:~$
```

3.2 Update the packages using the following command:
**sudo apt update**

```
labsuser@ip-172-31-40-202:~$ sudo apt update
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:6 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:7 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:8 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Hit:9 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
189 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

3.3 Execute the below command to install Nagios XI to extend the capabilities of Nagios Core:

**sudo curl https://assets.nagios.com/downloads/nagiosxi/install.sh | sudo sh**

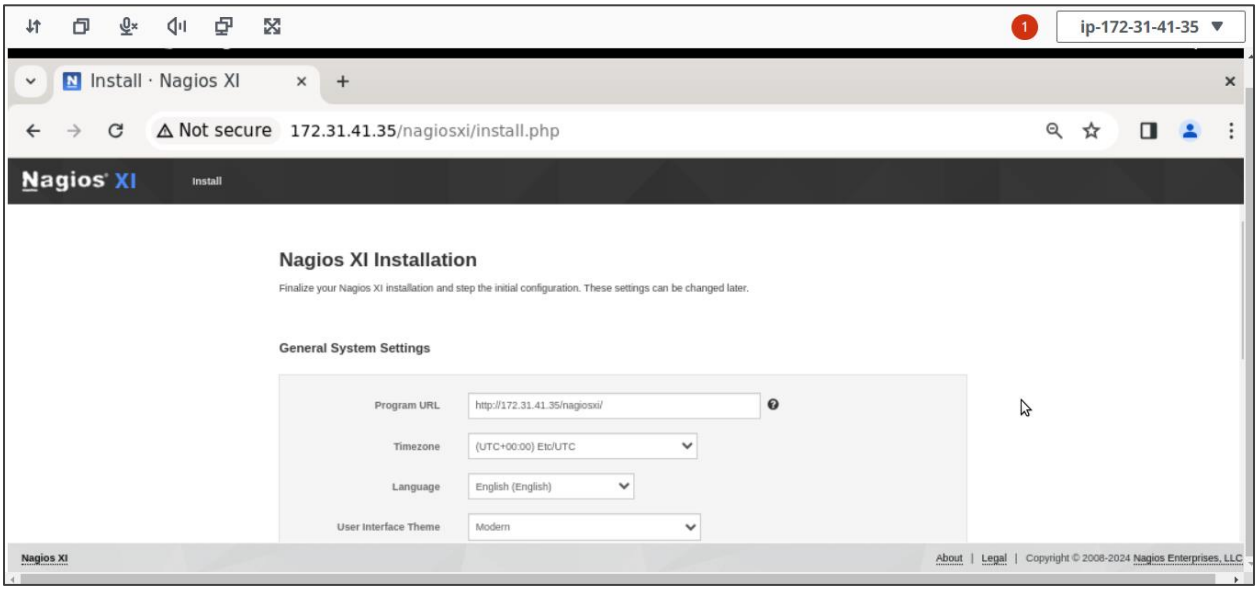
```
labsuser@ip-172-31-40-202:~$ sudo apt update
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:6 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:7 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:8 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Hit:9 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu jammy InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
189 packages can be upgraded. Run 'apt list --upgradable' to see them.
labsuser@ip-172-31-40-202:~$ sudo curl https://assets.nagios.com/downloads/nagiosxi/install.sh | sudo sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   629  100   629    0     0   2546      0 --:--:-- --:--:-- --:--:--  2556
/usr/bin/wget
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
wget is already the newest version (1.21.2-2ubuntu1).
0 upgraded, 0 newly installed, 0 to remove and 189 not upgraded.
check if /tmp/nagiosxi exists
Downloading latest Nagios XI release
--2024-02-15 19:40:52--  https://assets.nagios.com/downloads/nagiosxi/xi-latest.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 83805786 (80M) [application/x-gzip]
```

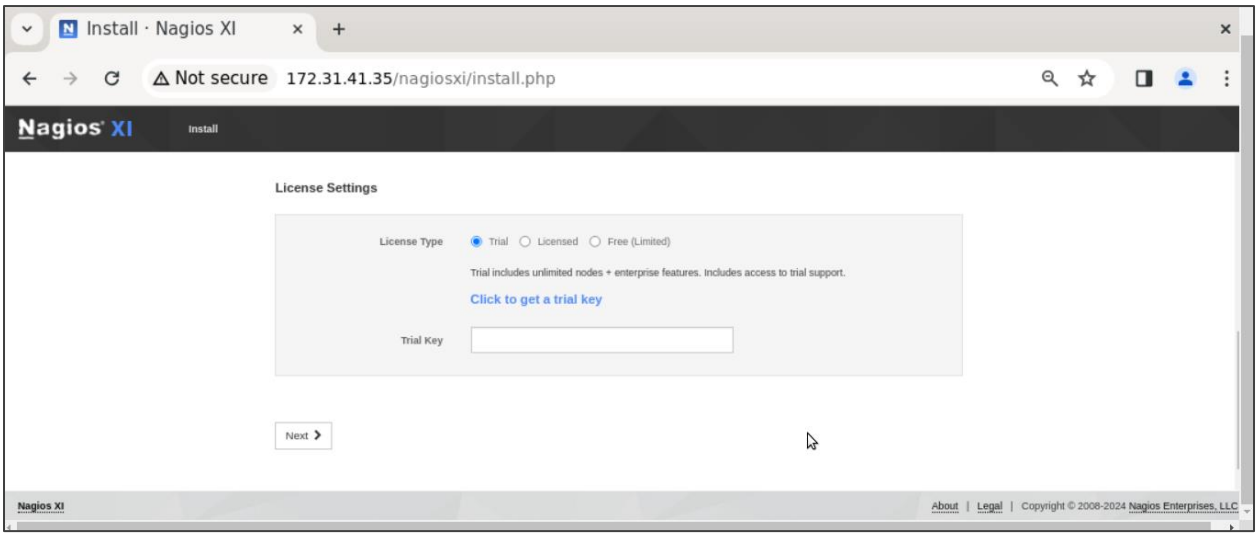
**Note**: The installation takes around 20 mins. Please wait till you get the **Installation Complete** message as shown below:

```
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
> Return Code: 0
-------------------------------------

CCM data imported OK.
RESULT=0
Running './F-startdaemons'...
Synchronizing state of mrtg.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable mrtg
Removed /etc/systemd/system/multi-user.target.wants/mrtg.service.
Daemons started OK
RESULT=0
Running './Z-webroot'...
RESULT=0

Nagios XI Installation Complete!
--------------------------------
```

3.4 Navigate to the user interface by using the URL provided in your terminal session as shown in the screenshot below:

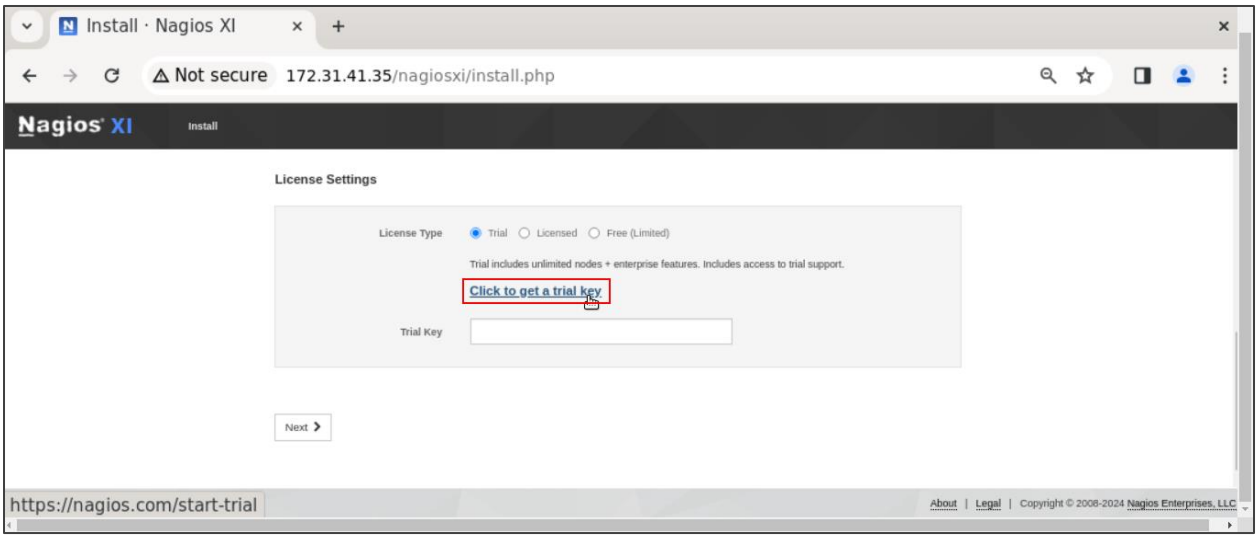
```
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
> Return Code: 0
-------------------------------------

CCM data imported OK.
RESULT=0
Running './F-startdaemons'...
Synchronizing state of mrtg.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable mrtg
Removed /etc/systemd/system/multi-user.target.wants/mrtg.service.
Daemons started OK
RESULT=0
Running './Z-webroot'...
RESULT=0

Nagios XI Installation Complete!
--------------------------------

You can access the Nagios XI web interface by visiting:
   http://172.31.41.35/nagiosxi/

labsuser@ip-172-31-41-35:~$
```



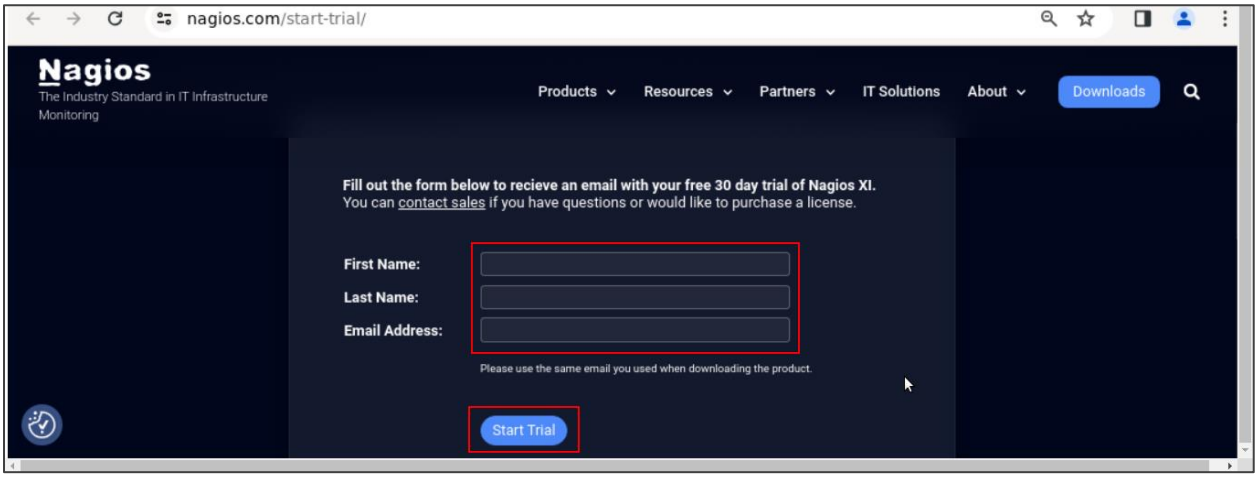



**Note:** In the General System Settings, make the changes as desired, choose your license settings; for demonstration, you may use the **Trial** version.

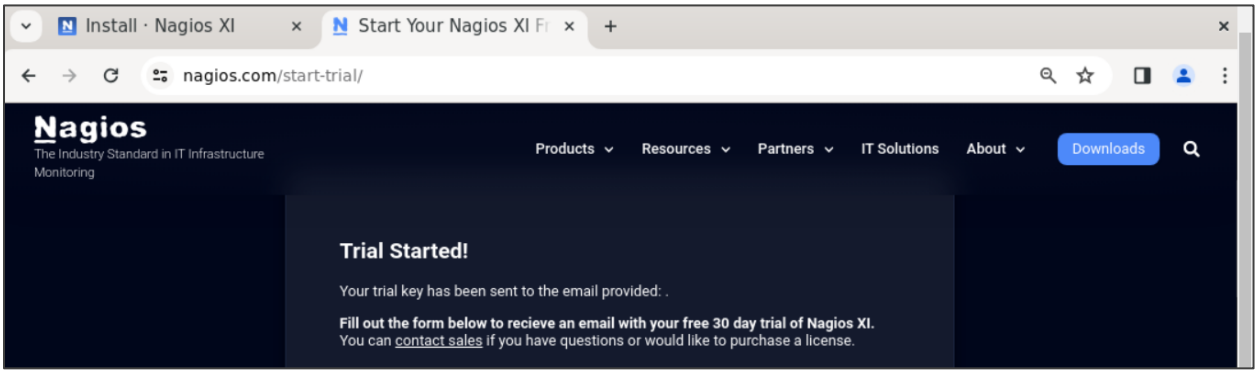
3.5 Click on the **Click to get a trial key** link as shown in the screenshot below:



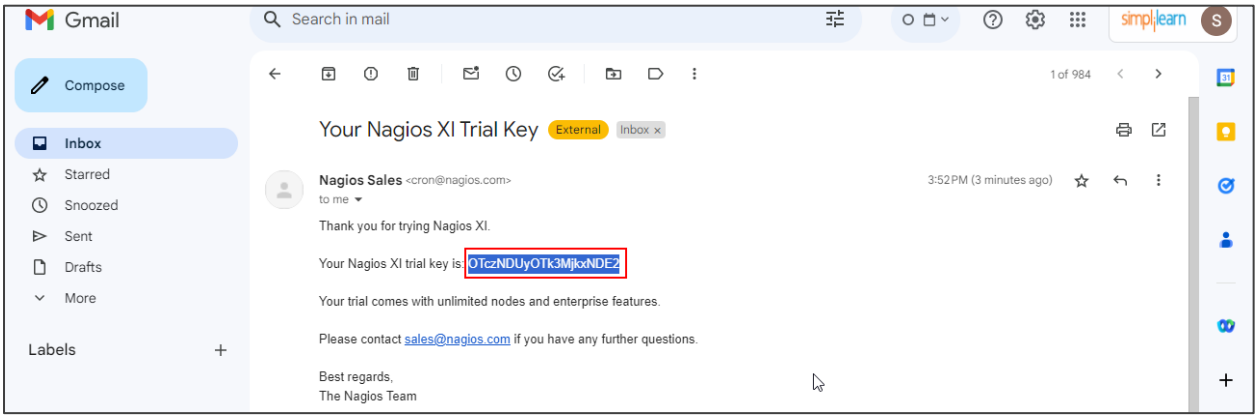
3.6 Add the required details, and then click on the **Start Trial** button as shown in the screenshot below:

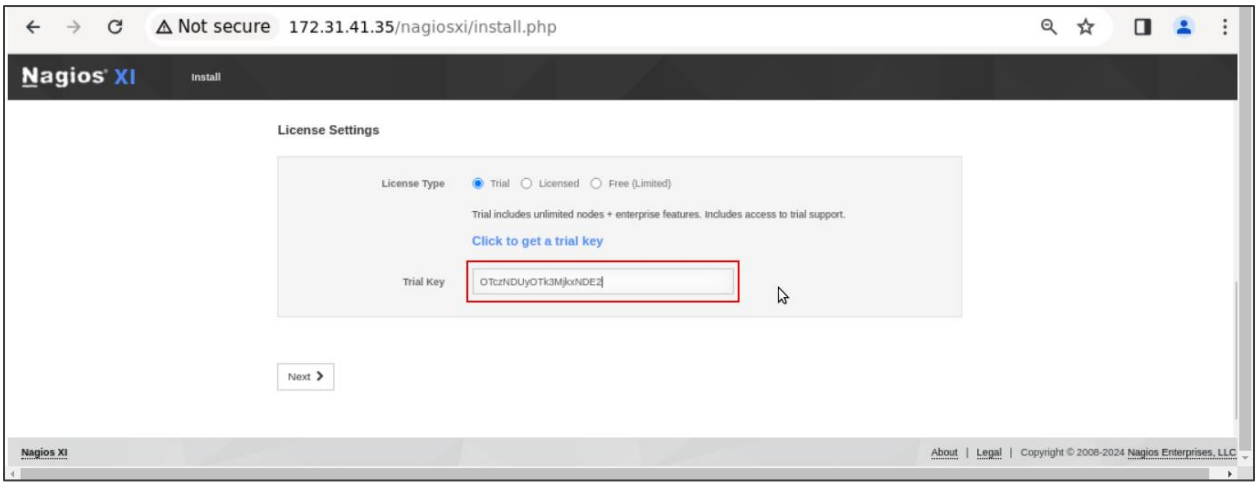



The trial key will be sent to your email.

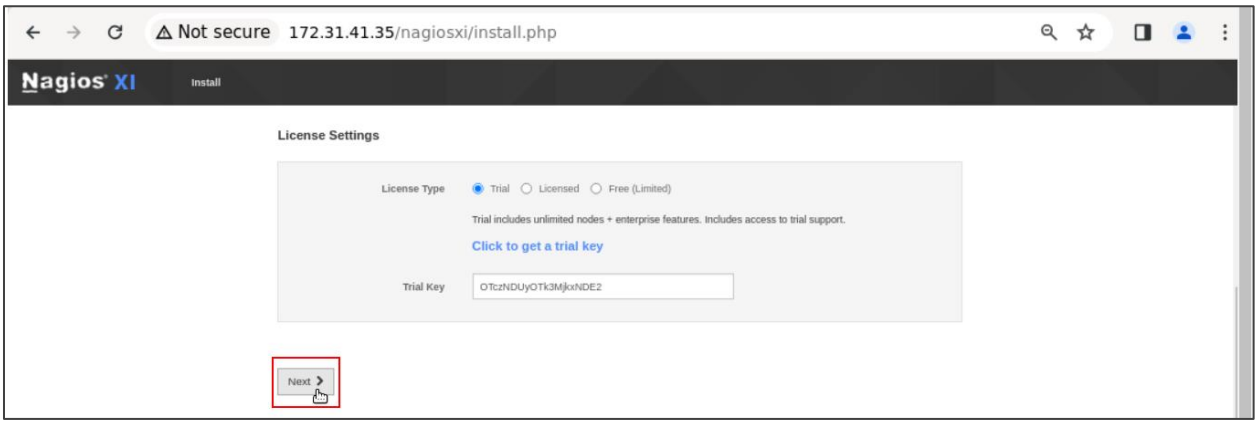



3.7 Navigate to the inbox of your email and copy the trial key and paste it in the **License Settings** as shown in the screenshot below:

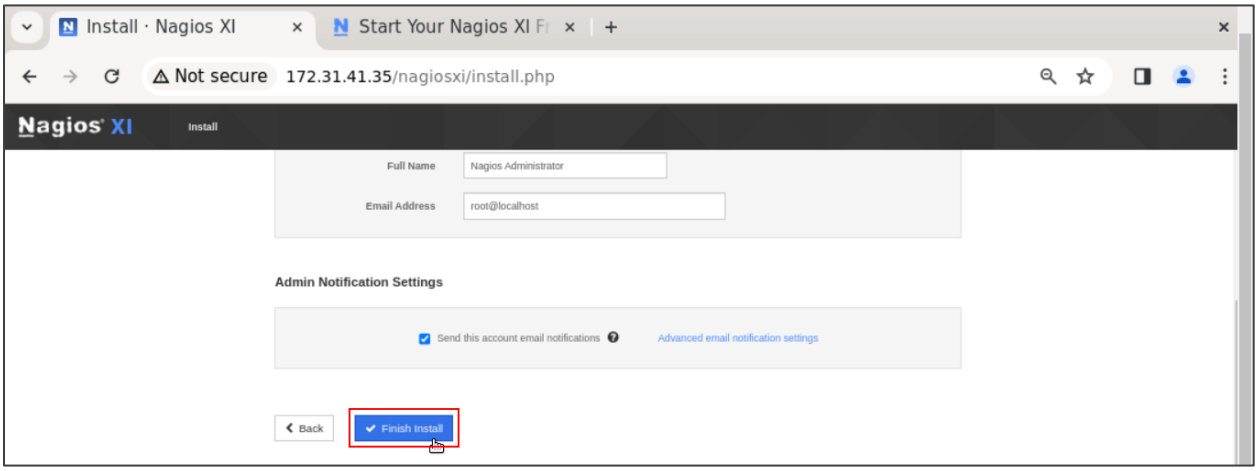





3.8 Click on the **Next** button to proceed

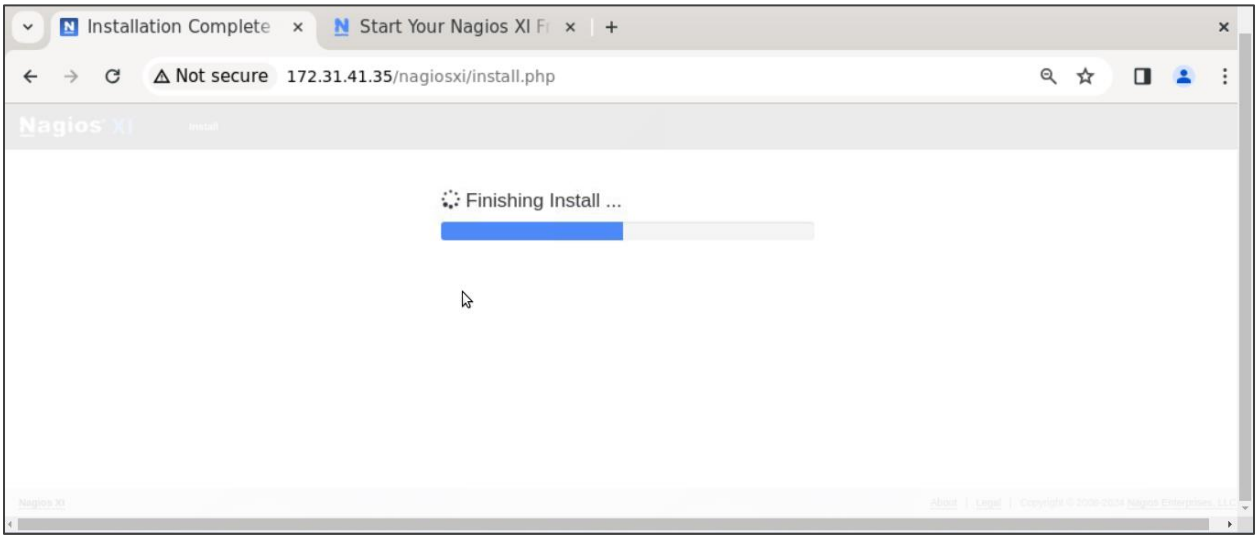



3.9 Scroll down and click on the **Finish Install** button as shown in the screenshot below:

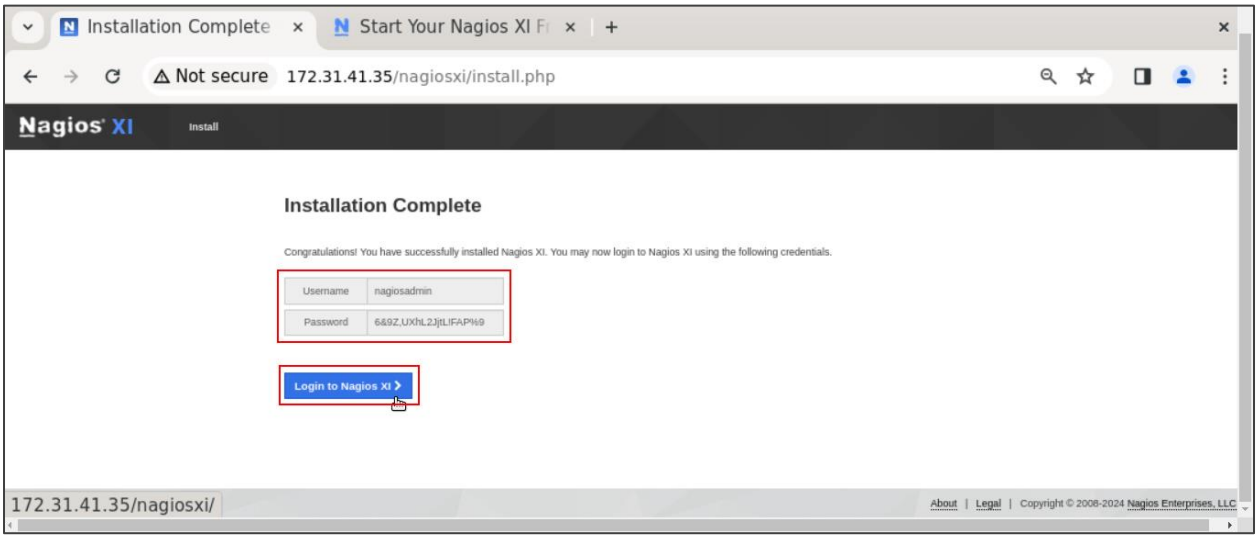


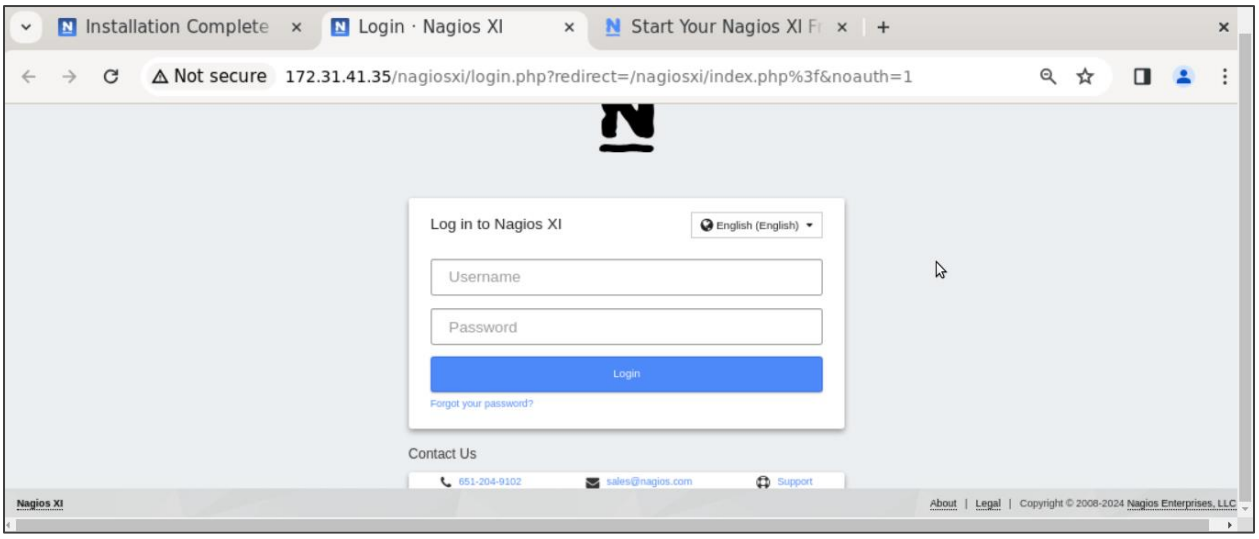
**Note:** In the **Admin Account Settings,** you may change the password as desired.

The installation takes around 2 minutes to complete as shown in the screenshot below:

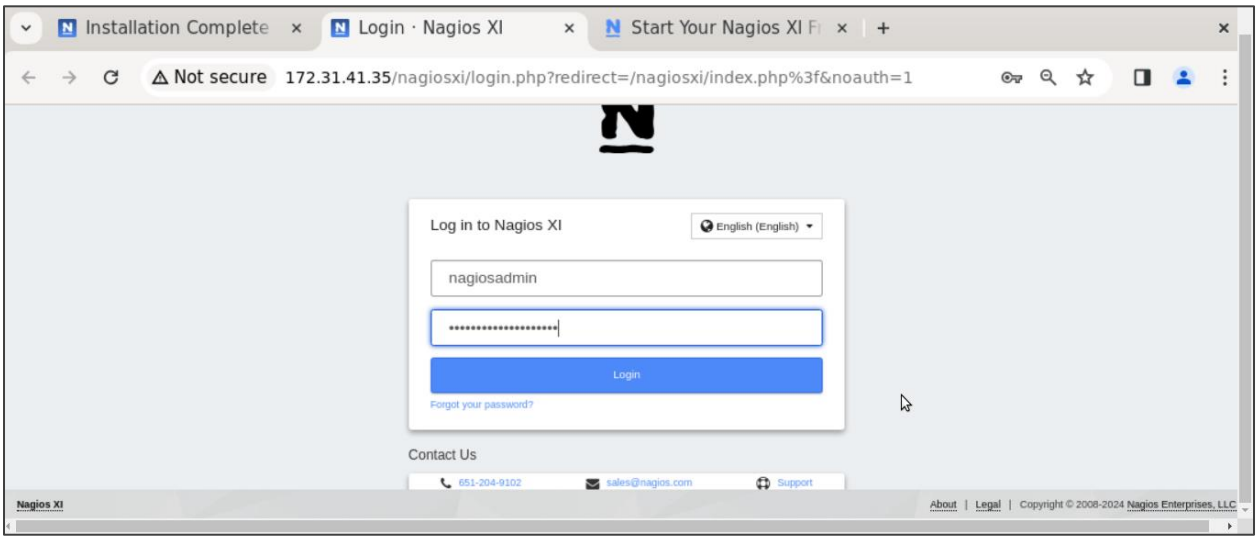



3.10 Copy the **Username** and **Password**, then click on the **Login to Nagios XI** button as shown in the screenshot below:

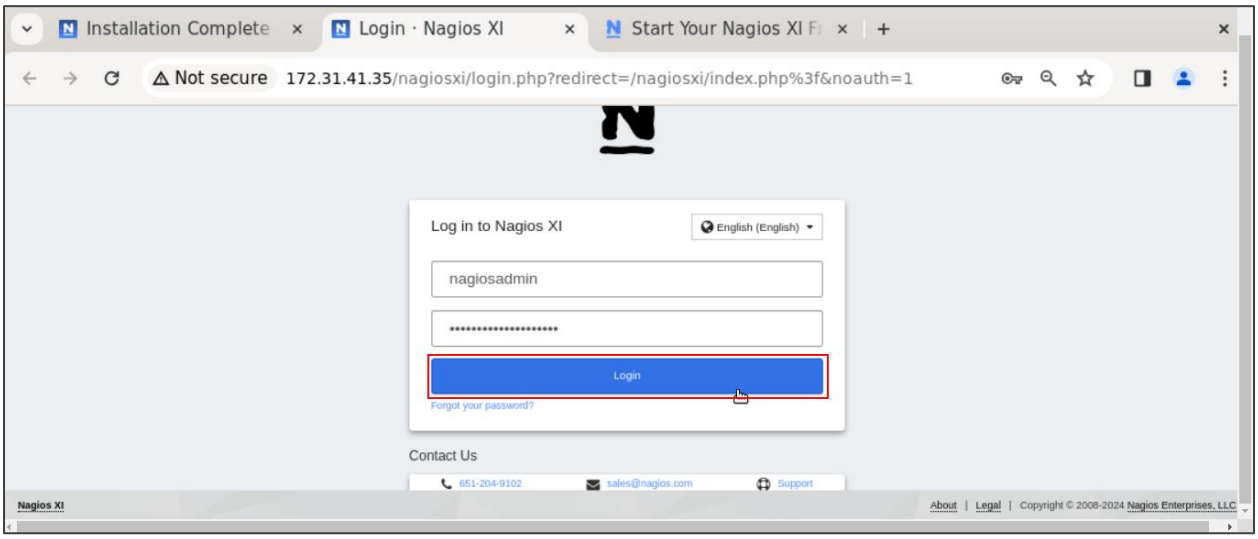




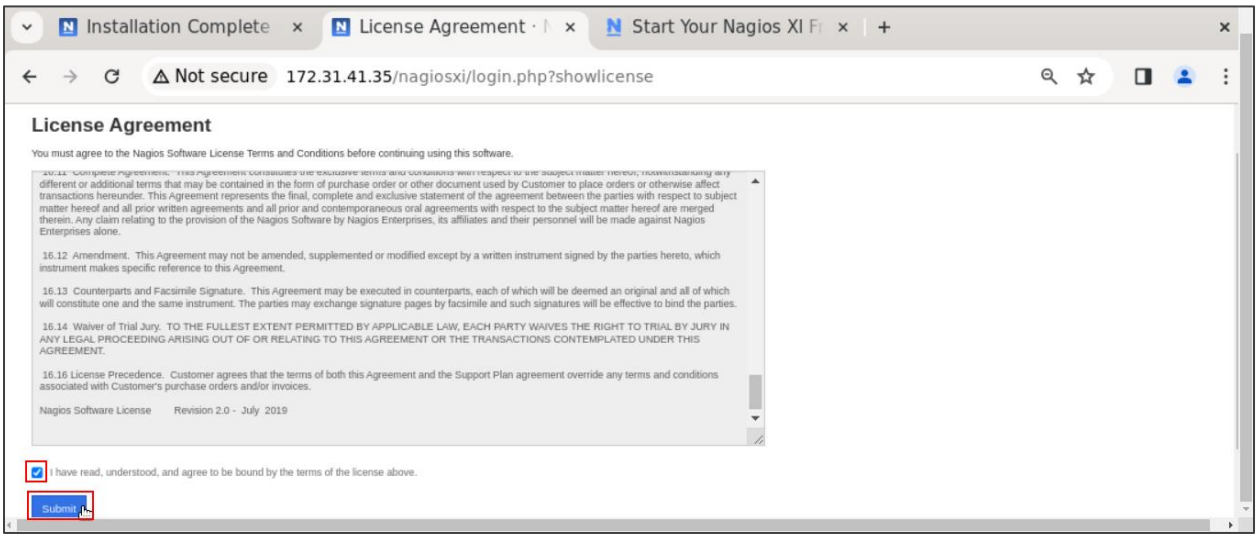

3.11 Add the copied **Username** and **Password** in the input field as shown in the screenshot below:



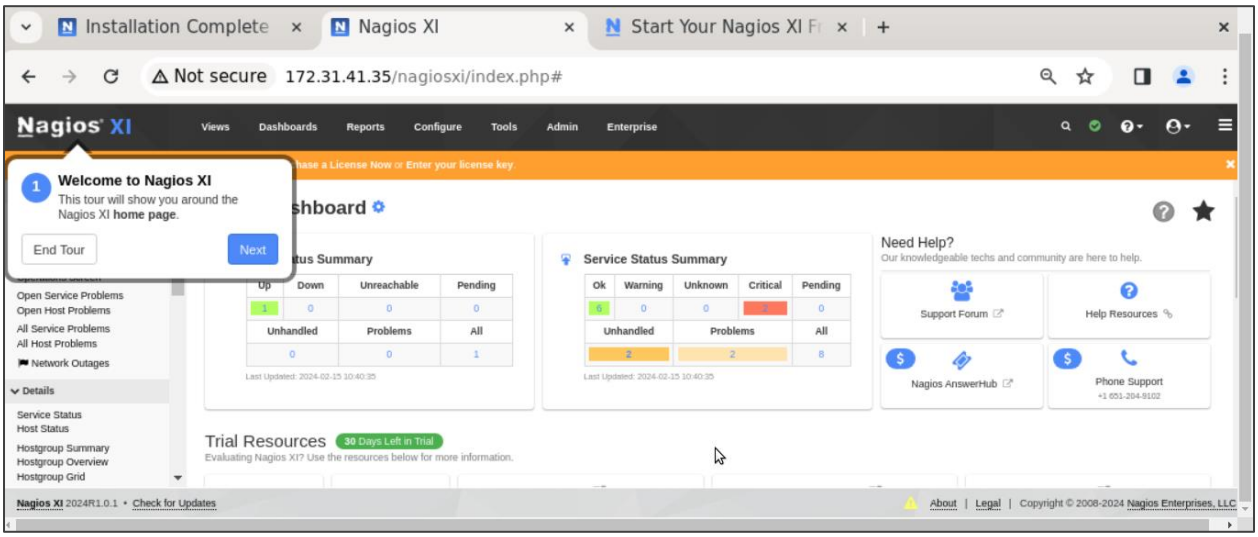
3.12 Click on the **Login** button




3.13 Scroll down, click on the **check box** to accept the License Agreement, and then click on the **Submit** button as shown in the screenshot below:



You will be logged into Nagios XI as shown in the screenshot below:




By following these steps, you have successfully installed and configured Nagios monitoring tool for efficiently managing system health and network status of your infrastructure.