

## **EXPERIMENT NO. 9**

**Aim:** To Understand Continuous monitoring and Installation and configuration of Nagios Core,

Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

### **STEPS TO INSTALL AND SETUP NAGIOS ON EC-2 INSTANCE**

1. Create an Amazon Linux EC-2 instance and select either existing key pair or create new

[EC2](#) > ... > Launch an instance

## Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

### Name and tags [Info](#)

Name

[Add additional tags](#)


### ▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below


Recents

Quick Start


Amazon Linux




macOS




Ubuntu




Windows




Red Hat



SUSE Li



  
Browse more AMIs  
Including AMIs from AWS, Marketplace and the Community

Instances (1) <a href="#">Info</a>									
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				Running	Last updated less than a minute ago <a href="#">Refresh</a> <a href="#">Connect</a> <a href="#">Instance state</a> <a href="#">Actions</a> <a href="#">Launch instances</a>				
<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input type="checkbox"/>	nagio-host	i-01ad97aa822f75271	Running	t2.micro	Initializing	<a href="#">View alarms</a>	us-east-1c	ec2-52-207-222-231.co...	52.207.222.231

2. Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere.



Now, we need to install necessary packages that are required to run nagios properly

## sudo yum install httpd php

```
Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64      apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64      generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.62-1.amzn2023.x86_64      httpd-core-2.4.62-1.amzn2023.x86_64      httpd-filesystem-2.4.62-1.amzn2023.noarch      httpd-tools-2.4.62-1.amzn2023.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64      libksft-1.1.34-5.amzn2023.0.2.x86_64      mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-2.0.27-1.amzn2023.0.3.x86_64      mod_lua-2.4.62-1.amzn2023.x86_64      nginx-filesystem-1.1.24-0-1.amzn2023.0.4.noarch      php8.3-8.3.10-1.amzn2023.0.1.x86_64
php8.3-cli-8.3.10-1.amzn2023.0.1.x86_64      php8.3-common-8.3.10-1.amzn2023.0.1.x86_64      php8.3-fpm-8.3.10-1.amzn2023.0.1.x86_64      php8.3-mbstring-8.3.10-1.amzn2023.0.1.x86_64
php8.3-opcache-8.3.10-1.amzn2023.0.1.x86_64      php8.3-pdo-8.3.10-1.amzn2023.0.1.x86_64      php8.3-process-8.3.10-1.amzn2023.0.1.x86_64      php8.3-sodium-8.3.10-1.amzn2023.0.1.x86_64
php8.3-xml-8.3.10-1.amzn2023.0.1.x86_64
```

## sudo yum install gcc glibc glibc-common

```
Installed:
annobin-docs-10.93-1.amzn2023.0.1.noarch      annobin-plugin-gcc-10.93-1.amzn2023.0.1.x86_64      c++-11.4.1-2.amzn2023.0.2.x86_64      gc-8.0.4-5.amzn2023.0.2.x86_64
gcc-11.4.1-2.amzn2023.0.2.x86_64      glibc-devel-2.34-52.amzn2023.0.11.x86_64      glibc-headers-x86-2.34-52.amzn2023.0.11.noarch      guile22-2.2.7-2.amzn2023.0.3.x86_64
kernel-headers-6.1.109-118.189.amzn2023.x86_64      libmpc-1.2.1-2.amzn2023.0.2.x86_64      libtool-ltdl-2.4.7-1.amzn2023.0.3.x86_64      libxcrypt-devel-4.4.33-7.amzn2023.x86_64
make-1:4.3-5.amzn2023.0.2.x86_64
```

## sudo yum install gd gd-devel

```
Installed:
brotli-1.0.9-4.amzn2023.0.2.x86_64      brotli-devel-1.0.9-4.amzn2023.0.2.x86_64      bzip2-devel-1.0.8-6.amzn2023.0.2.x86_64
cairo-1.17.6-2.amzn2023.0.1.x86_64      cmake-filesystem-3.22.2-1.amzn2023.0.4.x86_64      fontconfig-2.13.94-2.amzn2023.0.2.x86_64
fontconfig-devel-2.13.94-2.amzn2023.0.2.x86_64      fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch      freetype-2.13.2-5.amzn2023.0.3.x86_64
freetype-devel-2.13.2-5.amzn2023.0.3.x86_64      gd-2.3.3-5.amzn2023.0.3.x86_64      gd-devel-2.3.3-5.amzn2023.0.3.x86_64
glib2-devel-2.74.7-689.amzn2023.0.2.x86_64      google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch      google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
graphite2-1.3.14-7.amzn2023.0.2.x86_64      graphite2-devel-1.3.14-7.amzn2023.0.2.x86_64      harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64      harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64      jbigkit-libs-2.1-21.amzn2023.0.2.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch      libICE-1.0.10-6.amzn2023.0.2.x86_64      libSM-1.2.3-8.amzn2023.0.2.x86_64
libX11-1.7.2-3.amzn2023.0.4.x86_64      libX11-common-1.7.2-3.amzn2023.0.4.noarch      libX11-devel-1.7.2-3.amzn2023.0.4.x86_64
libX11-xcb-1.7.2-3.amzn2023.0.4.x86_64      libXau-1.0.9-6.amzn2023.0.3.x86_64      libXau-devel-1.0.9-6.amzn2023.0.2.x86_64
libXext-1.3.4-6.amzn2023.0.2.x86_64      libXpm-3.5.15-2.amzn2023.0.3.x86_64      libXpm-devel-3.5.15-2.amzn2023.0.3.x86_64
libXrender-0.9.10-14.amzn2023.0.2.x86_64      libXt-1.2.0-4.amzn2023.0.2.x86_64      libb1kld-devel-2.37.4-1.amzn2023.0.4.x86_64
libffi-devel-3.4.4-1.amzn2023.0.1.x86_64      libicu-67.1-7.amzn2023.0.3.x86_64      libicu-devel-67.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64      libjpeg-turbo-devel-2.1.4-2.amzn2023.0.5.x86_64      libmount-devel-2.37.4-1.amzn2023.0.4.x86_64
libpng-2.1.6-37-10.amzn2023.0.6.x86_64      libpng-devel-2.1.6-37-10.amzn2023.0.6.x86_64      libselinux-devel-3.4-5.amzn2023.0.2.x86_64
libsepol-devel-3.4-3.amzn2023.0.3.x86_64      libtiff-4.4.0-4.amzn2023.0.18.x86_64      libtiff-devel-4.4.0-4.amzn2023.0.18.x86_64
libwebp-1.2.4-1.amzn2023.0.6.x86_64      libwebp-devel-1.2.4-1.amzn2023.0.6.x86_64      libxcb-1.13.1-7.amzn2023.0.2.x86_64
libxcb-devel-1.13.1-7.amzn2023.0.2.x86_64      libxml2-devel-2.10.4-1.amzn2023.0.6.x86_64      pcre2-devel-10.40-1.amzn2023.0.3.x86_64
pcre2-utf16-10.40-1.amzn2023.0.3.x86_64      pcre2-utf32-10.40-1.amzn2023.0.3.x86_64      pixman-0.40.0-3.amzn2023.0.3.x86_64
sysprof-capture-devel-3.40.1-2.amzn2023.0.2.x86_64      xml-common-0.6.3-56.amzn2023.0.2.noarch      xorg-x11-proto-devel-2021.4-1.amzn2023.0.2.noarch
xz-devel-5.2.5-9.amzn2023.0.2.x86_64      zlib-devel-1.2.11-33.amzn2023.0.5.x86_64
```

Complete!

4. Create a new Nagios User with its password. You'll have to enter the password twice for confirmation. This is require as to give separate permissions for nagios

**sudo adduser -m nagios**

**sudo passwd nagios**

```
ip-172-31-33-179.ec2.internal ec2-user ~ sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
```

6. Create a new user group

```
sudo groupadd nagcmd
```

7. Use these commands so that you don't have to use sudo for Apache and Nagios

```
sudo usermod -a -G nagcmd nagios
```

```
sudo usermod -a -G nagcmd apache
```

```
ip-172-31-33-179.ec2.internal ec2-user ~ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
```

8. Create a new directory for Nagios downloads

```
mkdir ~/downloads
```

```
cd ~/downloads
```

9. Use wget to download the installation source zip files.

```
wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
```

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-10-04 13:40:59-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.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: 2065473 (2.0M) [application/x-gzip]
Saving to: 'nagios-4.5.5.tar.gz'

nagios-4.5.5.tar.gz      100%[=====>]
2024-10-04 13:41:00 (4.15 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/2065473]
```

```
wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
```

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-10-04 13:41:25-- https://nagios-plugins.org/download/nagios-plugins-2.4.11.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: 2753049 (2.6M) [application/x-gzip]
Saving to: 'nagios-plugins-2.4.11.tar.gz'

nagios-plugins-2.4.11.tar.gz 100%[=====>]
2024-10-04 13:41:26 (7.55 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]
```

10. Use tar to unzip the downloaded archive and change to that directory.

tar zxvf nagios-4.5.5.tar.gz

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > tar zxvf nagios-4.5.5.tar.gz
nagios-4.5.5/
nagios-4.5.5/.github/
nagios-4.5.5/.github/workflows/
nagios-4.5.5/.github/workflows/test.yml
nagios-4.5.5/.gitignore
nagios-4.5.5/CONTRIBUTING.md
nagios-4.5.5/Changelog
nagios-4.5.5/INSTALLING
nagios-4.5.5/LEGAL
nagios-4.5.5/LICENSE
nagios-4.5.5/Makefile.in
```

11. Navigate to the extracted folder and Run the configuration script with the same group name you previously created.

./configure --with-command-group=nagcmd

```
checking for Kerberos include files... configure: WARNING: could not find include files
checking for pkg-config... pkg-config
checking for SSL headers... configure: error: Cannot find ssl headers
```

We got error, because ssl headers library is not installed

It can be installed using **sudo yum install openssl-devel**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 > sudo yum install openssl-devel
Last metadata expiration check: 0:28:57 ago on Fri Oct 4 13:16:25 2024.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository        Size
=====
Installing:
openssl-devel                          x86_64            1:3.0.8-1.amzn2023.0.14  amazonlinux      3.0 M
Transaction Summary
-----
Install 1 Package
```

Now rerun

**./configure --with-command-group=nagcmd**

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

*** Configuration summary for nagios 4.5.5 2024-09-17 ***:

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/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): /usr/bin/traceroute

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

12. Compile the source code.

**make all**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/base'
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o ./nagios.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_wproc_list',
```

13. Install binaries, init script and sample config files. Lastly, set permissions on the external

command directory.

**sudo make install**

**sudo make install-init**

**sudo make install-config**

**sudo make install-commandmode**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/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 nagiosstats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/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/ec2-user/downloads/nagios-4.5.5/cgi'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
cd ./html && make install
```

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 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
```



```
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 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
```

```
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 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 ***
```

14. Edit the config file and change the email address so that we can receive timely alerts about the status of our system.

**sudo nano /usr/local/nagios/etc/objects/contacts.cfg**

```
define contact {
    contact_name      nagiosadmin          ; Short name of user
    use               generic-contact      ; Inherit default values from generic-contact template (defined above)
    alias             Nagios Admin         ; Full name of user
    email             2022.alok.yadav@ves.ac.in; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>
}

#####
```

And change email with your email

15. Configure the web interface. This is used to set up web server configuration of nagios dashboard.

**sudo make install-webconf**

```
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 sudo 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 ***
```

16. Create a nagiosadmin account for nagios login along with password. You'll have to specify the password twice.

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

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
```

17. Restart Apache

**sudo service httpd restart**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
```

18. Go back to the downloads folder and unzip the plugins zip file.

**cd ~/downloads**

**tar zxvf nagios-plugins-2.4.11.tar.gz**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-4.5.5 cd ~/downloads
tar zxvf nagios-plugins-2.4.11.tar.gz
nagios-plugins-2.4.11/
nagios-plugins-2.4.11/build-aux/
nagios-plugins-2.4.11/build-aux/compile
nagios-plugins-2.4.11/build-aux/config.guess
nagios-plugins-2.4.11/build-aux/config.rpath
nagios-plugins-2.4.11/build-aux/config.sub
nagios-plugins-2.4.11/build-aux/install-sh
nagios-plugins-2.4.11/build-aux/ltmain.sh
nagios-plugins-2.4.11/build-aux/missing
nagios-plugins-2.4.11/build-aux/mkinstalldirs
nagios-plugins-2.4.11/build-aux/depcomp
nagios-plugins-2.4.11/build-aux/snippet/
nagios-plugins-2.4.11/build-aux/snippet/_Noreturn.h
nagios-plugins-2.4.11/build-aux/snippet/arg-nonnull.h
```

19. Compile and install plugins

**cd nagios-plugins-2.4.11**

**./configure --with-nagios-user=nagios --with-nagios-group=nagios**

```
checking that generated files are newer than configure... done
configure: creating ./config.status
config.status: creating gl/Makefile
config.status: creating nagios-plugins.spec
config.status: creating tools/build_perl_modules
config.status: creating Makefile
config.status: creating tap/Makefile
config.status: creating lib/Makefile
config.status: creating plugins/Makefile
config.status: creating lib/tests/Makefile
config.status: creating plugins-root/Makefile
config.status: creating plugins-scripts/Makefile
config.status: creating plugins-scripts/utils.pm
config.status: creating plugins-scripts/utils.sh
config.status: creating perlmods/Makefile
config.status: creating test.pl
config.status: creating pkg/solaris/pkginfo
config.status: creating po/Makefile.in
config.status: creating config.h
config.status: config.h is unchanged
config.status: executing depfiles commands
config.status: executing libtool commands
config.status: executing po-directories commands
config.status: creating po/POTFILES
config.status: creating po/Makefile
```

**make**

**sudo make install**

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-plugins-2.4.11 sudo make install
Making install in gl
make[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/gl'
make install-recursive
make[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/gl'
make[3]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/gl'
make[4]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/gl'
if test yes = no; then \
  case 'linux-gnu' in \
    darwin[56]*) \
      need_charset_alias=true ;; \
    darwin* | cygwin* | mingw* | pw32* | cegcc*) \
      need_charset_alias=false ;; \
    *) \
      need_charset_alias=true ;; \
  esac ; \
else \
  need_charset_alias=false ; \
fi ; \
if $need_charset_alias; then \
  /bin/sh ../build-aux/mkinstalldirs /usr/local/nagios/lib ; \
fi ; \
```

20. Start Nagios and Add Nagios to the list of system services

**sudo chkconfig --add nagios**

## sudo chkconfig nagios on

```
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-plugins-2.4.11 sudo chkconfig --add nagios
error reading information on service nagios: No such file or directory
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-plugins-2.4.11 sudo chkconfig nagios on
Note: Forwarding request to 'systemctl enable nagios.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.
```

## Verify the sample configuration files

```
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
```

As we can see no errors were detected

## sudo service nagios start

```
# ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-plugins-2.4.11 sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
```

## 21. Check the status of Nagios

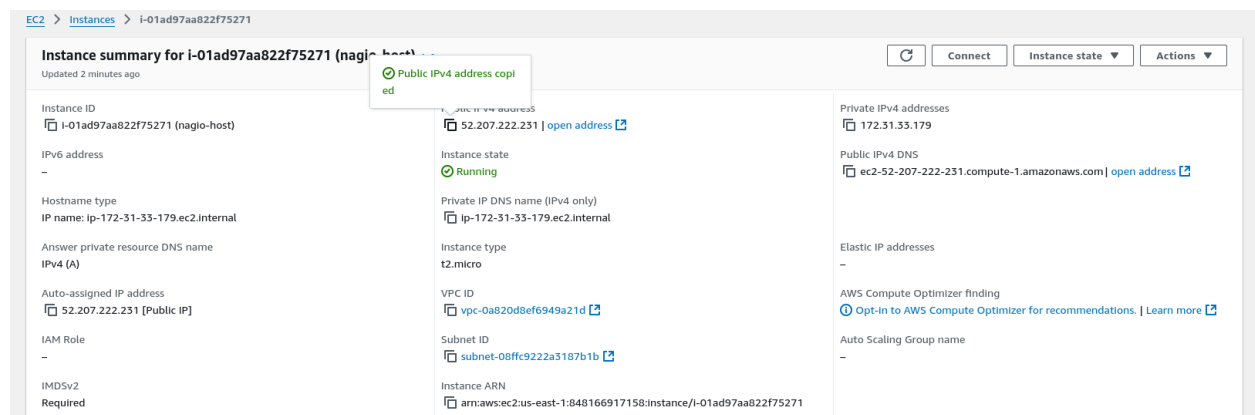
## sudo systemctl status nagios

```
ip-172-31-33-179.ec2.internal ec2-user ~ > downloads > nagios-plugins-2.4.11 sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Fri 2024-10-04 14:05:21 UTC; 33s ago
     Docs: https://www.nagios.org/documentation
  Process: 67432 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
  Process: 67433 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 67434 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 5.6M
      CPU: 67ms
   CGroup: /system.slice/nagios.service
           └─67434 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─67438 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─67439 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                 └─67440 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─67441 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                     └─67446 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: qh: core query handler registered
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: qh: echo service query handler registered
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: qh: help for the query handler registered
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: wproc: Successfully registered manager as @wproc with query handler
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: wproc: Registry request: name=Core Worker 67441;pid=67441
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: wproc: Registry request: name=Core Worker 67440;pid=67440
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: wproc: Registry request: name=Core Worker 67438;pid=67438
Oct 04 14:05:21 ip-172-31-33-179.ec2.internal nagios[67434]: wproc: Registry request: name=Core Worker 67439;pid=67439
Oct 04 14:05:22 ip-172-31-33-179.ec2.internal nagios[67434]: Successfully launched command file worker with pid 67446
```

The nagios service is running and working normally

22. Go back to EC2 Console and copy the Public IP address of this instance



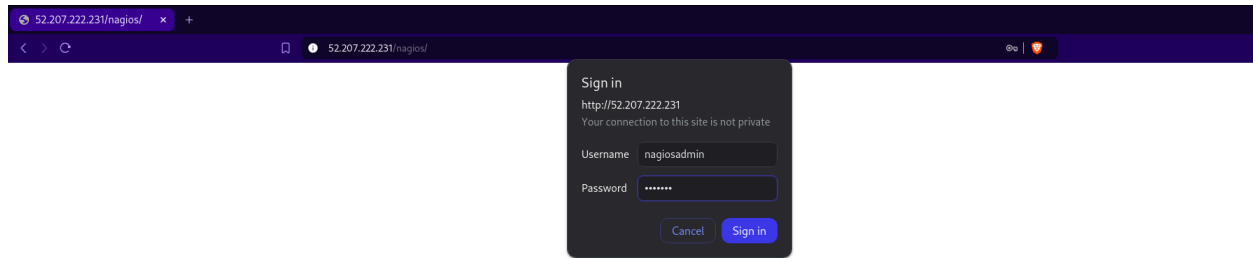
23. Open up your browser and look for `http://<your_public_ip_address>/nagios`

Enter username as nagiosadmin and password which we had set previously

Name: Alok Yadav  
No. 59

Div: D15C

Roll



24. After entering the correct credentials, you will see the Home page of Nagios.



This means that Nagios was correctly installed and configured with its plugins so far

## Now we can see system logs using nagios

Current Network Status

Last Updated: Fri Oct 4 14:20:53 UTC 2024

Updated every 90 seconds

Nagios® Core™ 4.5.5 - [www.nagios.org](http://www.nagios.org)

Logged in as [nagiosadmin](#)

View History For all hosts

View Notifications For All Hosts

View Host Status Detail For All Hosts

Host Status Totals

Up Down Unreachable Pending

1

0

0

0

All Problems

All Types

0

1

Service Status Totals

OK Warning Unknown Critical Pending

6

1

0

1

0

All Problems

All Types

2

8

Service Status Details For All Hosts

Limit Results:

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	10-04-2024 14:15:58	0d 0h 14m 55s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-04-2024 14:16:36	0d 0h 14m 17s	1/4	USERS OK - 1 users currently logged in
	HTTP	WARNING	10-04-2024 14:20:13	0d 0h 10m 40s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
	PING	OK	10-04-2024 14:17:51	0d 0h 13m 2s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
	Root Partition	OK	10-04-2024 14:18:28	0d 0h 12m 25s	1/4	DISK OK - free space: / 6031 MiB (74.31% inode=98%);
	SSH	OK	10-04-2024 14:19:06	0d 0h 11m 47s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
	Swap Usage	CRITICAL	10-04-2024 14:17:43	0d 0h 8m 10s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	10-04-2024 14:20:21	0d 0h 10m 32s	1/4	PROCS OK: 37 processes with STATE = RSZDT

Results 1 - 8 of 8 Matching Services

Above is the status of all the services running on Host Machine

### Conclusion:

We began the experiment by installing all the necessary packages required for Nagios. Next, we created a new user and group for Nagios, followed by installing the Nagios software through local compilation. Proper compilation is crucial to avoid errors during operation. After installation, we started both the `httpd` and Nagios services, enabling access to the Nagios dashboard, where critical system information is displayed.