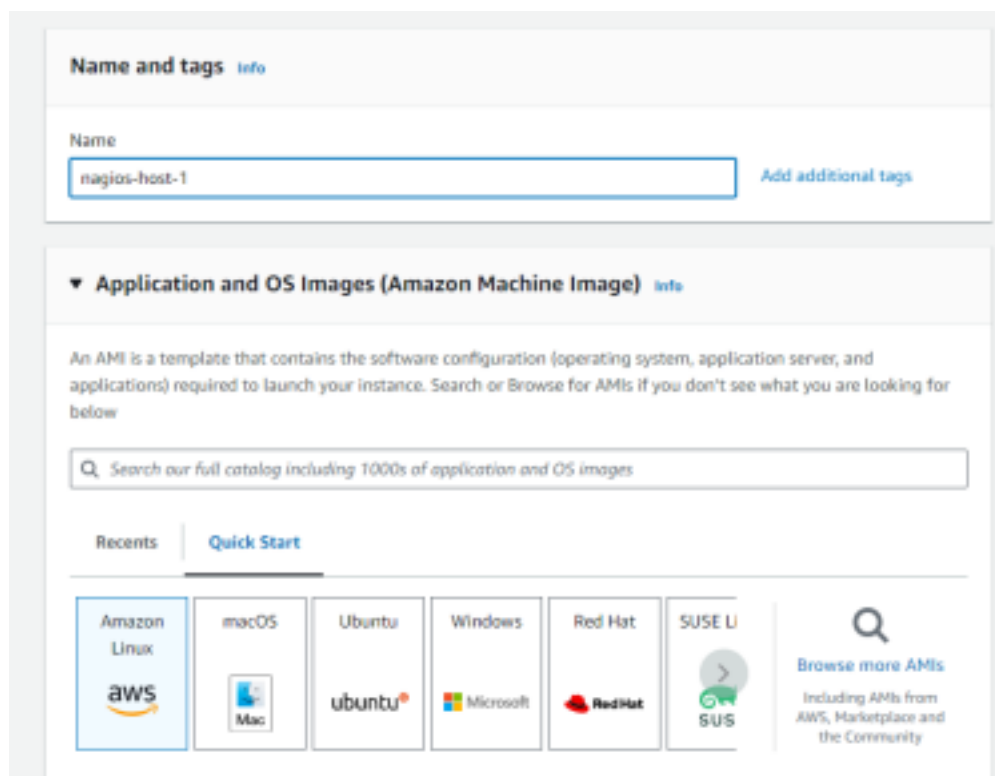


Experiment 9

Aim: To understand continuous monitoring and installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Steps:

1. Create an ec2 instance and select amazon Linux as the OS



2. Now we will create a key pair.

The screenshot shows the 'Create key pair' dialog box. It has a title bar 'Create key pair' with a close button. Below the title, there's a section 'Key pair name' with a text input field containing 'nagios_practical'. A note below says 'The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.' Below that is 'Key pair type' with two radio buttons: 'RSA' (selected) and 'ED25519'. Below that is 'Private key file format' with two radio buttons: '.pem' (selected) and '.ppk'. At the bottom, there's a yellow warning box with a triangle icon and text: 'When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)'. At the very bottom are 'Cancel' and 'Create key pair' buttons.

3. Use the created key pair while creating the instance

The screenshot shows the 'Key pair (login)' section in the AWS console. It has a title 'Key pair (login)' with an 'Info' link. Below the title, there's a paragraph: 'You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.' Below that is a label 'Key pair name - required' and a dropdown menu showing 'nagios_practical'. To the right of the dropdown is a 'Create new key pair' button with a plus icon.

4. Once the instance is successfully initiated go to security groups and select the security group id of the instance you just created.

(The security group name is visible during instance creation and also on the ec2 instances dashboard)

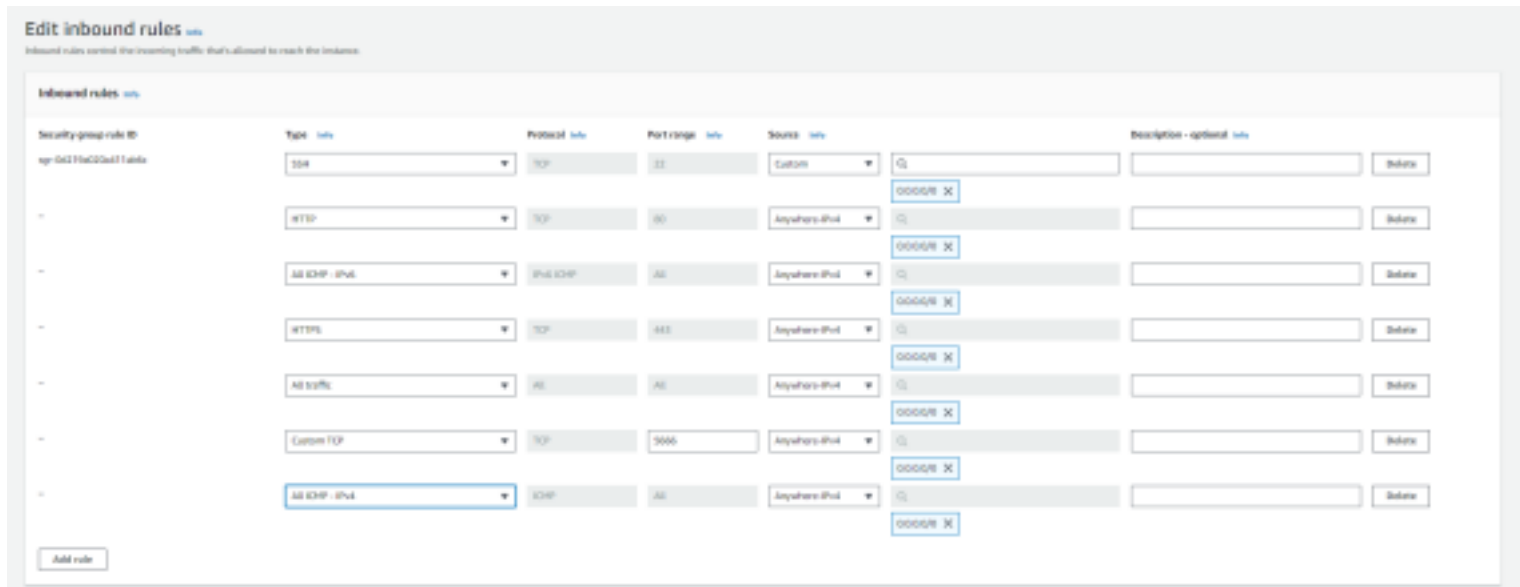
The screenshot shows the 'Security Groups' page in the AWS console. It has a title 'Security Groups (13)' with a 'Info' link. Below the title is a search bar 'Find resources by attribute or tag'. To the right are buttons for 'Actions', 'Export security groups to CSV', and 'Create security group'. Below these is a table with columns: 'Name', 'Security group ID', 'Security group name', 'VPC ID', 'Description', and 'Owner'. The table contains 13 rows of security groups.

Name	Security group ID	Security group name	VPC ID	Description	Owner
--	sg-08e6b8b11e36203	launch-wizard-18	vpc-b6138811:88117f9c02	launch-wizard-18 created 2024-09-08...	025866268342
--	sg-03a0887c17123b4477	launch-wizard-21	vpc-b6138811:88117f9c02	launch-wizard-21 created 2024-09-11...	025866268342
--	sg-5273187c1279d36	launch-wizard-14	vpc-b6138811:88117f9c02	launch-wizard-14 created 2024-09-08...	025866268342
--	sg-0308773648d494e0d	launch-wizard-32	vpc-b6138811:88117f9c02	launch-wizard-32 created 2024-10-01...	025866268342
--	sg-0b4f1390a630465c	launch-wizard-15	vpc-b6138811:88117f9c02	launch-wizard-15 created 2024-09-08...	025866268342
--	sg-077c7908ba112b2b6	default	vpc-b6138811:88117f9c02	default VPC security group	025866268342
--	sg-00a9d34946f13866a2	launch-wizard-19	vpc-b6138811:88117f9c02	launch-wizard-19 created 2024-06-29...	025866268342

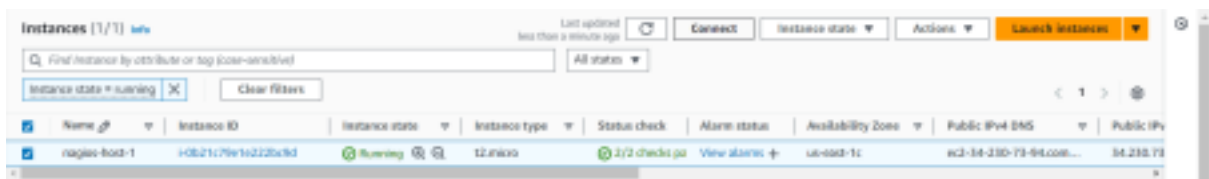
5. To edit the inbound rules select the "Edit inbound rules" button



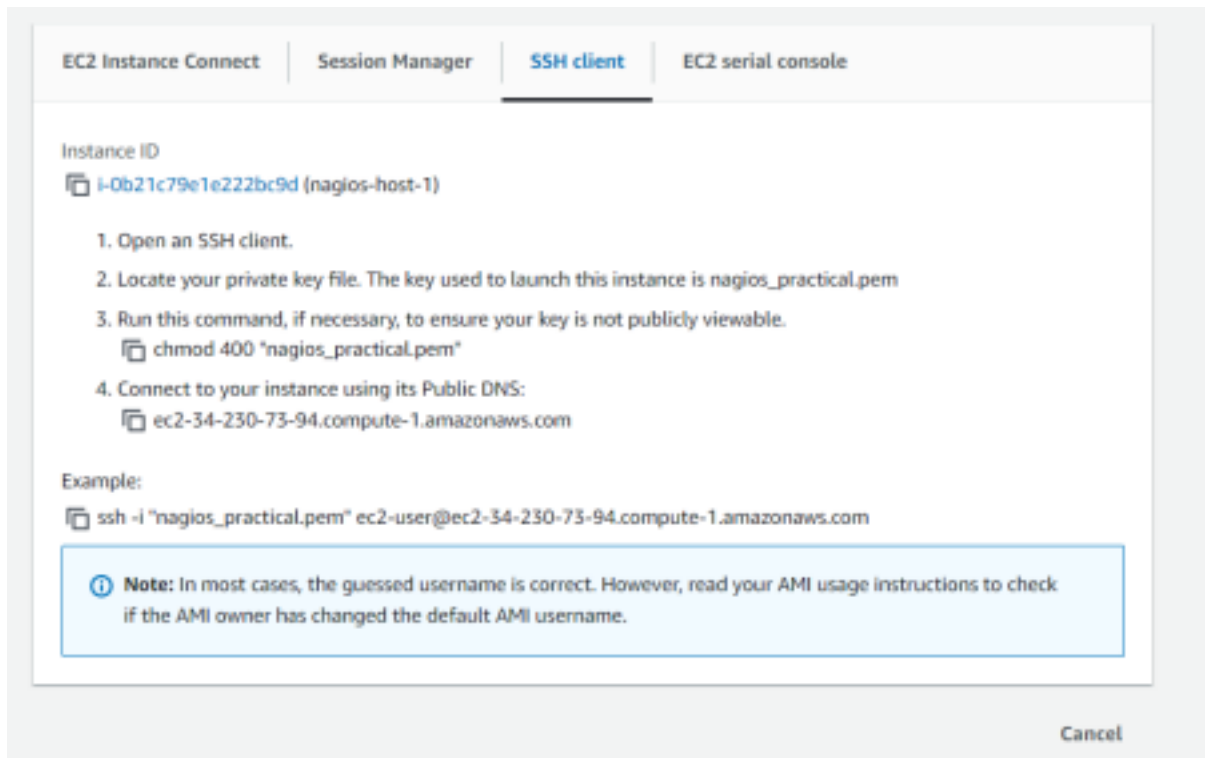
6. Add the rules as given in the screenshot below



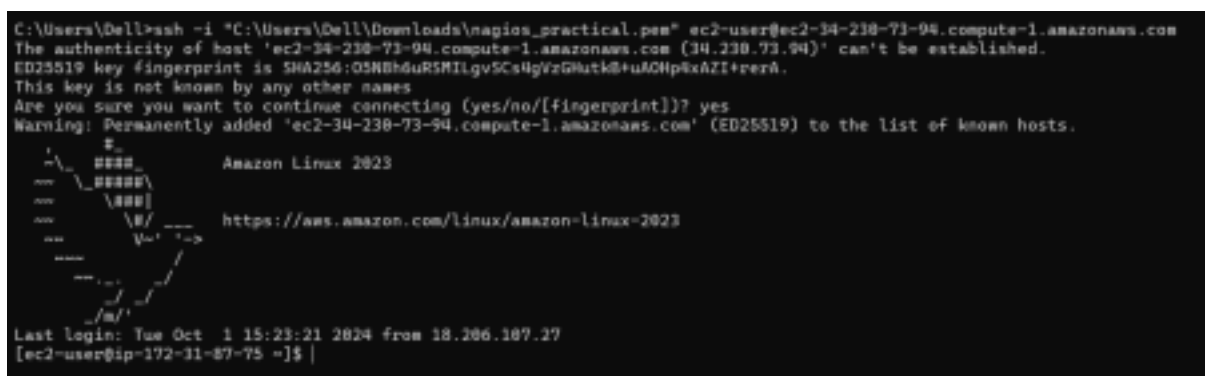
7. Connect the instance



8. Copy the ssh command given in the ssh client section.

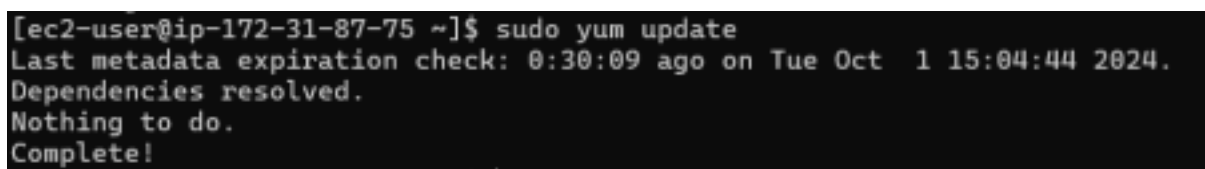


9. In your terminal paste the copied command, just replace the .pem file name with the actual location where the .pem file is downloaded in your system



10. Now install the following packages using yum:

```
sudo yum update
```



```
sudo yum install httpd php
```

```
[ec2-user@ip-172-31-07-75 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:31:22 ago on Tue Oct 1 15:04:04 2024.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
=====
Installing:
httpd                                 x86_64            2.4.62-1.amzn2023 amazonlinux       48 k
php8.3                               x86_64            8.3.10-1.amzn2021 amazonlinux       10 k
Installing dependencies:
apr                                   x86_64            1.7.2-2.amzn2023 amazonlinux       129 k
apr-util                             x86_64            1.6.3-1.amzn2023 amazonlinux       98 k
generic-logos-httpd                 noarch            18.0.0-12.amzn2023 amazonlinux       19 k
httpd-core                           x86_64            2.4.62-1.amzn2023 amazonlinux       1.4 M
httpd-filesystem                     noarch            2.4.62-1.amzn2023 amazonlinux       18 k
httpd-tools                          x86_64            2.4.62-1.amzn2023 amazonlinux       81 k
libbrotli                             x86_64            1.0.9-4.amzn2023 amazonlinux       315 k
libsodium                             x86_64            1.0.19-4.amzn2023 amazonlinux       176 k
libsqlite                             x86_64            3.45.1-1.amzn2023 amazonlinux       241 k
mailcap                              noarch            2.1.49-3.amzn2021 amazonlinux       11 k
nghttp-filesystem                   noarch            1:1.24.0-1.amzn2023 amazonlinux       9.8 k
php8.3-cli                           x86_64            8.3.10-1.amzn2021 amazonlinux       2.7 M
php8.3-common                        x86_64            8.3.10-1.amzn2021 amazonlinux       737 k
php8.3-process                       x86_64            8.3.10-1.amzn2021 amazonlinux       45 k
php8.3-sql                           x86_64            8.3.10-1.amzn2021 amazonlinux       134 k
Installing weak dependencies:
apr-util-openssl                     x86_64            1.6.3-1.amzn2023 amazonlinux       17 k
mod_http2                           x86_64            2.0.27-1.amzn2023 amazonlinux       166 k
mod_lua                             x86_64            2.4.62-1.amzn2023 amazonlinux       61 k
php8.3-fpm                          x86_64            8.3.10-1.amzn2023 amazonlinux       1.9 M
php8.3-embedding                     x86_64            8.3.10-1.amzn2023 amazonlinux       528 k
php8.3-opcache                       x86_64            8.3.10-1.amzn2023 amazonlinux       379 k
php8.3-pdo                           x86_64            8.3.10-1.amzn2023 amazonlinux       89 k
php8.3-sodium                        x86_64            8.3.10-1.amzn2023 amazonlinux       41 k
Transaction Summary
=====
Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.62-1.amzn2023.noarch
libbrotli-1.0.9-4.amzn2023.x86_64
mod_http2-2.0.27-1.amzn2023.0.1.x86_64
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-pdo-8.3.10-1.amzn2023.0.1.x86_64
php8.3-process-8.3.10-1.amzn2023.0.1.x86_64
php8.3-sql-8.3.10-1.amzn2023.0.1.x86_64
php8.3-sodium-8.3.10-1.amzn2023.0.1.x86_64
php8.3-embedding-8.3.10-1.amzn2023.0.1.x86_64
php8.3-opcache-8.3.10-1.amzn2023.0.1.x86_64
php8.3-fpm-8.3.10-1.amzn2023.0.1.x86_64
php8.3-embedding-8.3.10-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-core-2.4.62-1.amzn2023.x86_64
libbrotli-1.0.9-4.amzn2023.0.1.x86_64
mailcap-2.1.49-3.amzn2021.0.3.noarch
nghttp-filesystem-1:1.24.0-1.amzn2023.0.4.noarch
php8.3-common-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
Complete!
```

sudo yum install gcc glibc glibc-common

```
[ec2-user@ip-172-31-07-75 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:31:03 ago on Tue Oct 1 15:04:00 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
=====
Installing:
gcc                                   x86_64            11.4.1-2.amzn2023 amazonlinux       32 M
Installing dependencies:
annobin-docs                         noarch            10.93-1.amzn2023 amazonlinux       92 k
annobin-plugin-gcc                   x86_64            10.93-1.amzn2023 amazonlinux       807 k
c++                                  x86_64            11.4.1-2.amzn2023 amazonlinux       18 M
c++-libs                             x86_64            8.0.4-5.amzn2023 amazonlinux       105 k
glibc-devel                           x86_64            2.34-52.amzn2023 amazonlinux       27 k
glibc-headers-x86                     noarch            2.34-52.amzn2023 amazonlinux       427 k
guile22                               x86_64            2.2.7-2.amzn2023 amazonlinux       6.4 M
kernel-headers                       x86_64            6.1.109-118.109.amzn2023 amazonlinux       1.0 M
libgcc                                x86_64            1.2.2-2.amzn2023 amazonlinux       62 k
libtool-ltdl                          x86_64            2.4.7-1.amzn2023 amazonlinux       28 k
libxcrypt-devel                       x86_64            4.4.33-7.amzn2023 amazonlinux       32 k
make                                  x86_64            1:4.3-5.amzn2023 amazonlinux       536 k
Transaction Summary
=====
Install 13 Packages

Installed:
annobin-docs-10.93-1.amzn2023.0.1.noarch
gcc-11.4.1-2.amzn2023.0.2.x86_64
glibc-headers-x86-2.34-52.amzn2023.0.11.noarch
libmpc-1.2.1-2.amzn2023.0.2.x86_64
make-1.4.3-5.amzn2023.0.2.x86_64
annobin-plugin-gcc-10.93-1.amzn2023.0.1.x86_64
gcc-11.4.1-2.amzn2023.0.2.x86_64
guile22-2.2.7-2.amzn2023.0.3.x86_64
libtool-ltdl-2.4.7-1.amzn2023.0.3.x86_64
libxcrypt-devel-4.4.33-7.amzn2023.x86_64
c++-11.4.1-2.amzn2023.0.2.x86_64
glibc-devel-2.34-52.amzn2023.0.11.x86_64
kernel-headers-6.1.109-118.109.amzn2023.x86_64
libxcrypt-devel-4.4.33-7.amzn2023.x86_64
Complete!
```

sudo yum install gd gd-devel

```
[ec2-user@ip-172-31-87-75 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:38:32 ago on Tue Oct 3 15:04:00 2023.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
-----
Installing:
gd                                     x86_64            2.3.3-5.amzn2023.0.3  amazonlinux      109 k
gd-devel                             x86_64            2.3.3-5.amzn2023.0.3  amazonlinux      38 k
Installing dependencies:
brotli                               x86_64            1.0.9-4.amzn2023.0.2  amazonlinux      310 k
brotli-devel                         x86_64            1.0.9-4.amzn2023.0.2  amazonlinux      31 k
bzip2-devel                         x86_64            1.0.8-6.amzn2023.0.2  amazonlinux      210 k
cairo                               x86_64            1.17.6-2.amzn2023.0.1  amazonlinux      680 k
cmake-filesystem                    x86_64            3.22.2-3.amzn2023.0.4  amazonlinux      16 k
fontconfig                          x86_64            2.13.94-2.amzn2023.0.2  amazonlinux      773 k
fontconfig-devel                   x86_64            2.13.94-2.amzn2023.0.2  amazonlinux      128 k
fonts-filesystem                   noarch            1:2.0.5-12.amzn2023.0.2  amazonlinux      9.5 k
freetype                           x86_64            2.13.2-5.amzn2023.0.1  amazonlinux      423 k
freetype-devel                     x86_64            2.13.2-5.amzn2023.0.1  amazonlinux      912 k
glib2-devel                        x86_64            2.74.7-699.amzn2023.0.2  amazonlinux      486 k
google-noto-fonts-common           noarch            20201206-2.amzn2023.0.2  amazonlinux      15 k
google-noto-sans-vf-fonts          noarch            20201206-2.amzn2023.0.2  amazonlinux      492 k
graphite2                          x86_64            1.3.14-7.amzn2023.0.2  amazonlinux      97 k
graphite2-devel                   x86_64            1.3.14-7.amzn2023.0.2  amazonlinux      71 k
harfbuzz                          x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      800 k
harfbuzz-devel                    x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      880 k
harfbuzz-icu                      x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      18 k
jbigkit-liba                       x86_64            2.1-21.amzn2023.0.3  amazonlinux      56 k
langpacks-core-font-en             noarch            3.0-21.amzn2023.0.4  amazonlinux      10 k

Installed:
brotli-1.0.9-4.amzn2023.0.2.x86_64
bzip2-devel-1.0.8-6.amzn2023.0.2.x86_64
cmake-filesystem-3.22.2-3.amzn2023.0.4.x86_64
fontconfig-devel-2.13.94-2.amzn2023.0.2.x86_64
fontconfig-2.13.2-5.amzn2023.0.1.x86_64
freetype-2.13.2-5.amzn2023.0.1.x86_64
gd-2.3.3-5.amzn2023.0.3.x86_64
glib2-devel-2.74.7-699.amzn2023.0.2.x86_64
google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
graphite2-devel-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64
jbigkit-liba-2.1-21.amzn2023.0.3.x86_64
libICE-1.0.10-6.amzn2023.0.2.x86_64
libX11-1.7.2-3.amzn2023.0.4.x86_64
libX11-devel-1.7.2-3.amzn2023.0.4.x86_64
libXaa-1.0.9-6.amzn2023.0.2.x86_64
libXext-1.3.4-6.amzn2023.0.2.x86_64
libXpm-devel-3.5.15-2.amzn2023.0.3.x86_64
libXt-2.0.4-4.amzn2023.0.2.x86_64
libffi-devel-3.4.4-1.amzn2023.0.1.x86_64
libica-devel-0.7.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-devel-2.1.4-2.amzn2023.0.5.x86_64
libpng-2.1.6.37-10.amzn2023.0.6.x86_64
libselinux-devel-3.4-5.amzn2023.0.2.x86_64
libtiff-4.0.4-6.amzn2023.0.18.x86_64
libwebp-1.2.4-1.amzn2023.0.6.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
libxml2-devel-2.10.4-1.amzn2023.0.6.x86_64
pcr2-atf10-10.40-1.amzn2023.0.3.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64
xml-common-0.6.3-66.amzn2023.0.2.noarch
xz-devel-5.2.5-9.amzn2023.0.2.x86_64

brotli-devel-1.0.9-4.amzn2023.0.2.x86_64
fontconfig-2.13.94-2.amzn2023.0.2.x86_64
fontconfig-devel-2.13.94-2.amzn2023.0.2.noarch
freetype-devel-2.13.2-5.amzn2023.0.1.x86_64
gd-devel-2.3.3-5.amzn2023.0.3.x86_64
google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
graphite2-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
lib59-1.2.3-8.amzn2023.0.2.x86_64
libX11-common-1.7.2-3.amzn2023.0.4.noarch
libX11-xcb-1.7.2-3.amzn2023.0.4.x86_64
libXaa-devel-1.0.9-6.amzn2023.0.2.x86_64
libXpm-3.5.15-2.amzn2023.0.3.x86_64
libXrender-0.9.10-14.amzn2023.0.2.x86_64
libblkid-devel-2.37.4-1.amzn2023.0.4.x86_64
libica-0.7.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libseant-devel-2.37.4-1.amzn2023.0.4.x86_64
libpng-devel-2.1.6.37-10.amzn2023.0.6.x86_64
libsepol-devel-3.4-3.amzn2023.0.3.x86_64
libtiff-devel-4.0.4-6.amzn2023.0.18.x86_64
libwebp-devel-1.2.4-1.amzn2023.0.6.x86_64
libxcb-devel-1.13.1-7.amzn2023.0.2.x86_64
pcr2-devel-10.40-1.amzn2023.0.3.x86_64
pcr2-atf32-10.40-1.amzn2023.0.3.x86_64
sysprep-capture-devel-3.00.1-2.amzn2023.0.2.x86_64
xorg-x11-proto-devel-2021.0-1.amzn2023.0.2.noarch
zlib-devel-1.2.11-33.amzn2023.0.5.x86_64

Complete!
[ec2-user@ip-172-31-87-75 ~]$
```

11. Create a new Nagios User with its password using the below given commands.

```
sudo adduser -m nagios
sudo passwd nagios
```

```
[ec2-user@ip-172-31-87-75 ~]$ sudo adduser -m nagios
sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-87-75 ~]$
```

12. Create a new user group

```
sudo groupadd nagcmd
```

```
[ec2-user@ip-172-31-87-75 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-87-75 ~]$
```

13. Next execute 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
```

```
[ec2-user@ip-172-31-87-75 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-87-75 ~]$ |
```

14. Create a new directory for Nagios downloads

```
mkdir ~/downloads
```

```
cd ~/downloads
```

```
[ec2-user@ip-172-31-87-75 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-87-75 downloads]$ |
```

15. Use wget to download the source zip files.

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

```
[ec2-user@ip-172-31-87-75 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-10-01 15:55:47-- 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:fe77:45ce
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%[=====] 1.97M  5.54MB/s  in 0.4s
2024-10-01 15:55:47 (5.54 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/2065473]
[ec2-user@ip-172-31-87-75 downloads]$ |
```

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

```
[ec2-user@ip-172-31-87-75 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-10-01 15:57:19-- 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%[=====] 2.62M  4.14MB/s  in 0.6s
2024-10-01 15:57:20 (4.14 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]
```

16. Use tar to unzip and change to that directory.

```
tar zxvf nagios-4.5.5.tar.gz
```

```
[ec2-user@ip-172-31-87-75 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/LLEGAL
nagios-4.5.5/LICENSE
nagios-4.5.5/Makefile.in
nagios-4.5.5/README.md
nagios-4.5.5/THANKS
nagios-4.5.5/UPGRADING
nagios-4.5.5/aclocal.m4
nagios-4.5.5/autoconf-macros/
nagios-4.5.5/autoconf-macros/.gitignore
nagios-4.5.5/autoconf-macros/CHANGELOG.md
nagios-4.5.5/autoconf-macros/LICENSE
nagios-4.5.5/autoconf-macros/LICENSE.md
nagios-4.5.5/autoconf-macros/README.md
nagios-4.5.5/autoconf-macros/add_group_user
nagios-4.5.5/autoconf-macros/ax_nagios_get_distrib
nagios-4.5.5/autoconf-macros/ax_nagios_get_files
```

17. We have to now change the directory to nagios-4.5.5, for this first verify whether nagios-4.5.5 exists by using ls command.

```
[ec2-user@ip-172-31-87-75 downloads]$ ls
nagios-4.5.5  nagios-4.5.5.tar.gz  nagios-plugins-2.4.11.tar.gz
```

18. As nagios-4.5.5 is present we will now use cd command to change directory.

```
[ec2-user@ip-172-31-87-75 downloads]$ cd nagios-4.5.5
[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ |
```

19. Now we will install openssl dev library by using the command:
sudo yum install openssl-devel

```
Transaction Summary
-----
Install 1 Package

Total download size: 3.0 M
Installed size: 4.7 M
Is this ok [y/N]: y
Downloading Packages:
openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64.rpm                                23 MB/s | 3.0 MB  00:00
-----
Total                                                                           18 MB/s | 3.0 MB  00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing                :
  Installing                : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1
  Running scriptlet        : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1
  Verifying                 : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1

Installed:
openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64
```

20. Run the configuration script

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


```
[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ ./configure --with-command-group=nagcmd
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 the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for stdio.h... yes
checking for stdlib.h... yes
checking for string.h... yes
```

*** 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.

```
[ec2-user@ip-172-31-87-75 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 nebmodes.o nebmodes.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',
    inlined from 'get_worker' at workers.c:277:12:
workers.c:253:17: warning: '%s' directive argument is null [-Wformat-overflow=]
   253 |         log_debug_info(DEBUGL_CHECKS, 1, "Found specialized worker(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
       |         ^
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o events.o events.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o flapping.o flapping.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o logging.o logging.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o macros-base.o ../common/macros.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o netutils.o netutils.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o notifications.o notifications.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o sehandlers.o sehandlers.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o utils.o utils.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o retention-base.o ./sretention.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o xretention-base.o ../xdata/xrddefault.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o comments-base.o ../common/comments.c
```

22. To install binaries, init script and sample config files run

sudo make install

sudo make install-init

sudo make install-config

sudo make install-commandmode

```
[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode
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
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/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
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images/logos
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/includes
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/ssi
/usr/bin/install -c -m 664 -o nagios -g nagios ./robots.txt /usr/local/nagios/share
/usr/bin/install -c -m 664 -o nagios -g nagios ./jquery.html /usr/local/nagios/share
rm -f /usr/local/nagios/share/index.html
rm -f /usr/local/nagios/share/main.html
rm -f /usr/local/nagios/share/side.html
```

23. In the config file edit the email address

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

```
#####
#
# CONTACTS
#
#####

# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.

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             d2022.nayaab.jindani@ves.ac.in ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****

}
```

24. To configure the web interface run:

```
sudo make install-webconf
```

```
[ec2-user@ip-172-31-87-75 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 ***

[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ |
```

25. Create a nagios admin account and password

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

```
[ec2-user@ip-172-31-87-75 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
```

26. Restart apache

```
sudo service httpd restart
```

```
[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-87-75 nagios-4.5.5]$ |
```

27. Go back to the downloads folder by using “cd ~/downloads” and unzip the plugins zip file using

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

```
[ec2-user@ip-172-31-87-75 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
nagios-plugins-2.4.11/build-aux/snippet/c++defs.h
nagios-plugins-2.4.11/build-aux/snippet/warn-on-use.h
nagios-plugins-2.4.11/build-aux/test-driver
nagios-plugins-2.4.11/config_test/
nagios-plugins-2.4.11/config_test/Makefile
nagios-plugins-2.4.11/config_test/run_tests
nagios-plugins-2.4.11/config_test/child_test.c
nagios-plugins-2.4.11/gl/
```

28. Compile and install the plugins

```
cd nagios-plugins-2.0.3
```

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

```
[ec2-user@ip-172-31-87-75 downloads]$ cd nagios-plugins-2.4.11
./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 make supports nested variables... yes
checking whether to enable maintainer-specific portions of Makefiles... no
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
```

make

sudo make install

```
[ec2-user@ip-172-31-87-75 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 ; \
if test -f /usr/local/nagios/lib/charset.alias; then \
  sed -f ref-add.sed /usr/local/nagios/lib/charset.alias > /usr/local/nagios/lib/charset.tmp ; \
  /usr/bin/install -c -o nagios -g nagios -m 644 /usr/local/nagios/lib/charset.tmp /usr/local/nagios/lib/charset.alias ; \
  rm -f /usr/local/nagios/lib/charset.tmp ; \
else \
  if $need_charset_alias; then \
    sed -f ref-add.sed charset.alias > /usr/local/nagios/lib/charset.tmp ; \
    /usr/bin/install -c -o nagios -g nagios -m 644 /usr/local/nagios/lib/charset.tmp /usr/local/nagios/lib/charset.alias ; \
```

29. Run below given commands to start nagios:

sudo chkconfig --add nagios

sudo chkconfig nagios on

```
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ sudo chkconfig --add nagios
sudo chkconfig nagios on
error reading information on service nagios: No such file or directory
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.
```

sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagi
Nagios Core 4.5.5
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-09-17
License: GPL

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

Running pre-flight check on configuration data...

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

Total Warnings: 0
Total Errors: 0

Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ |
```

If the message says no errors detected then run “sudo service nagios start”

```
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ sudo service nagios s
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ |
```

sudo systemctl status nagios


```

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
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-87-75 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 Tue 2024-10-01 16:32:27 UTC; 48s ago
     Docs: https://www.nagios.org/documentation
  Process: 66684 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
  Process: 66693 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
 Main PID: 66694 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 5.7M
      CPU: 82ms
  CGroup: /system.slice/nagios.service
          └─66694 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─66695 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                └─66696 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─66697 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                      └─66698 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                         └─66739 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: qh: core query handler registered
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: qh: echo service query handler registered
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: qh: help for the query handler registered
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: wproc: Successfully registered manager as @wproc with query
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: wproc: Registry request: name=Core Worker 66696;pid=66696
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: wproc: Registry request: name=Core Worker 66695;pid=66695
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: wproc: Registry request: name=Core Worker 66697;pid=66697
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: wproc: Registry request: name=Core Worker 66698;pid=66698
Oct 01 16:32:27 ip-172-31-87-75.ec2.internal nagios[66694]: Successfully launched command file worker with pid 66739
[ec2-user@ip-172-31-87-75 nagios-plugins-2.4.11]$

```

We can see that Nagios has been initialized correctly and its status is active.

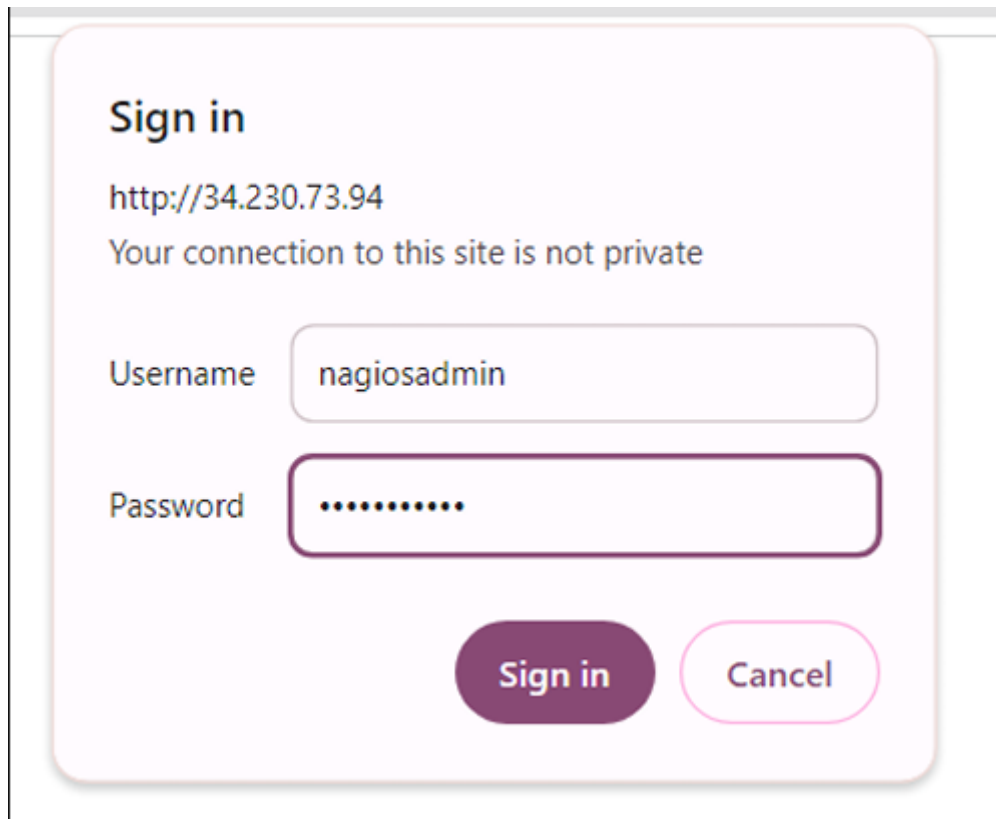
30. Go back to your instances and copy the public IPv4 address

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input checked="" type="checkbox"/>	nagios-host-1	i-0b21c79e1e222bc9d	Running	t2.micro	2/2 checks pa	View alarms +	us-east-1c	ec2-34-230-73-94.com

i-0b21c79e1e222bc9d (nagios-host-1)	
Details Status and alarms Monitoring Security Networking Storage Tags	
Instance summary Info Instance ID i-0b21c79e1e222bc9d (nagios-host-1) IPv6 address - Hostname type IP name: ip-172-31-87-75.ec2.internal	Public IPv4 address 34.230.73.94 open address Instance state Running Private IP DNS name (IPv4 only) ip-172-31-87-75.ec2.internal
	Private IPv4 addresses 172.31.87.75 Public IPv4 DNS ec2-34-230-73-94.compute-1.amazonaws.com

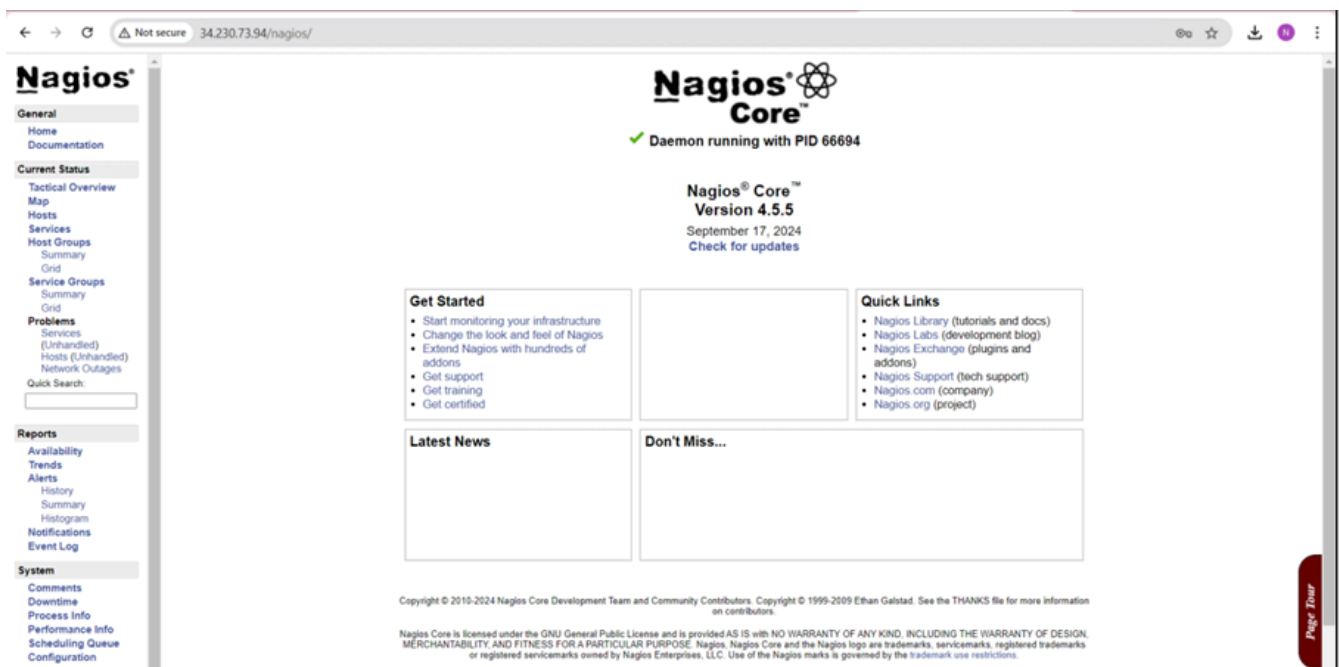
31. Lastly, go to your web browser and type “<http://<public-IPv4-address>/nagios>”
 Replace public-IPv4-address with the public ip address of your instance which you copied.

You will get a prompt to enter the username and password that have been set for nagios admin in step 25.



A sign-in dialog box for Nagios Core. It has a title 'Sign in' and shows the URL 'http://34.230.73.94'. A warning message states 'Your connection to this site is not private'. There are two input fields: 'Username' with the value 'nagiosadmin' and 'Password' with masked characters. At the bottom are two buttons: 'Sign in' and 'Cancel'.

You will see the below shown page after entering credentials.



The Nagios Core 4.5.5 dashboard is displayed in a web browser. The left sidebar contains navigation links for General, Current Status, Problems, Reports, and System. The main content area shows the Nagios Core logo, version 4.5.5, and the date September 17, 2024. It also indicates that the daemon is running with PID 66694. Below this, there are sections for 'Get Started', 'Quick Links', 'Latest News', and 'Don't Miss...'. The 'Get Started' section lists steps like monitoring infrastructure, changing look and feel, and getting support. The 'Quick Links' section provides links to Nagios Library, Nagios Labs, Nagios Exchange, Nagios Support, Nagios.com, and Nagios.org. The 'Latest News' and 'Don't Miss...' sections are currently empty. At the bottom, there is a copyright notice and a disclaimer.

Conclusion: In this experiment, the primary challenge I encountered was accessing the Nagios web interface due to a "Forbidden: You do not have permission to access this resource" error. This issue was resolved by modifying the inbound security rules and verifying that all necessary files were installed in the correct directories. Additionally, it is crucial to restart Apache after making any changes to ensure they take effect. Once Nagios was activated without any errors, the output was successfully displayed on the web interface.