Experiment - 9

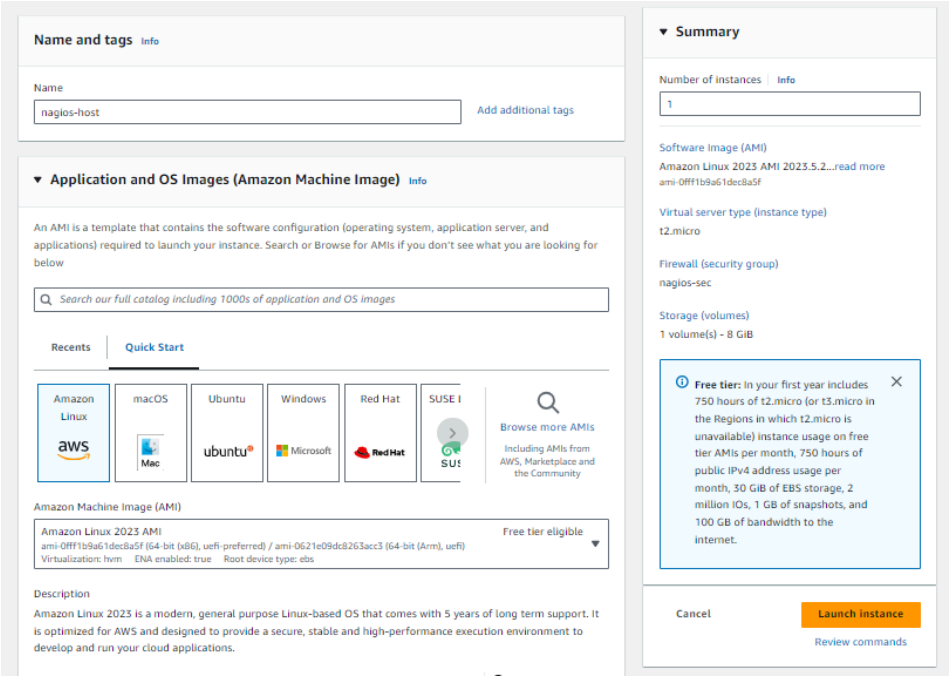
Aim: To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

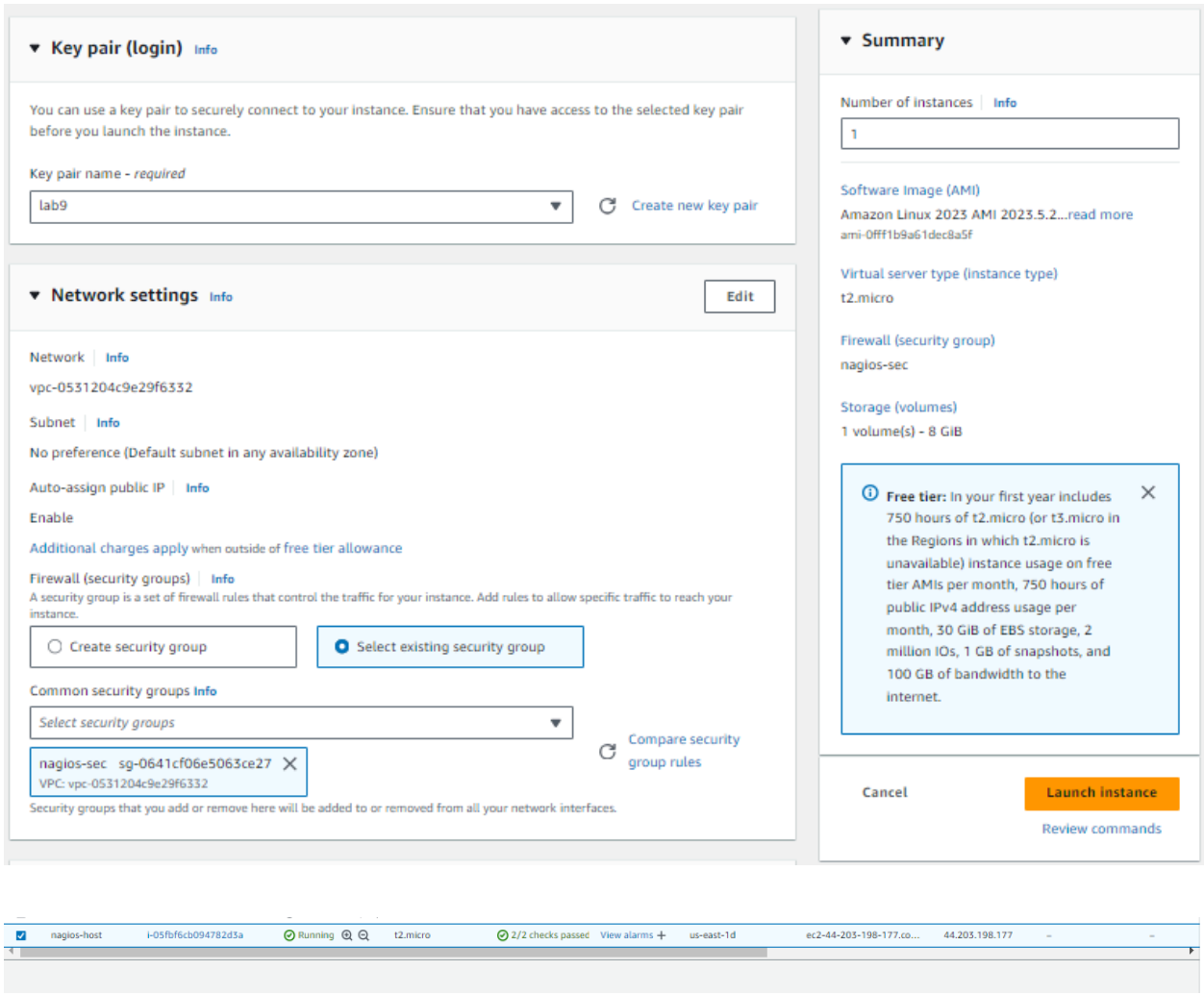
Theory:

What is Nagios?
Nagios is an open-source software for continuous monitoring of systems, networks, and infrastructures. It runs plugins stored on a server that is connected with a host or another server on your network or the Internet. In case of any failure, Nagios alerts about the issues so that the technical team can perform the recovery process immediately.
Nagios is used for continuous monitoring of systems, applications, service and business processes in a DevOps culture
.
Installation of Nagios

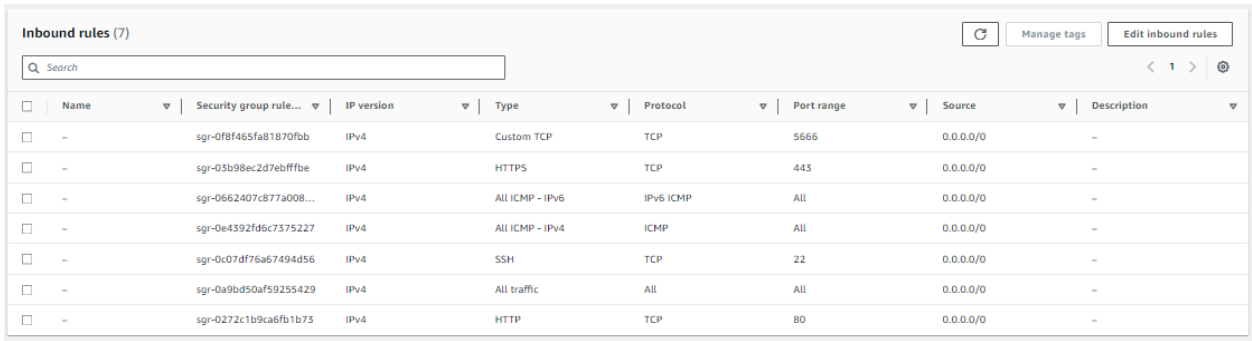
Prerequisites: AWS Free Tier

Steps:
1. Create an Amazon Linux EC2 Instance in AWS and name it - nagios-host



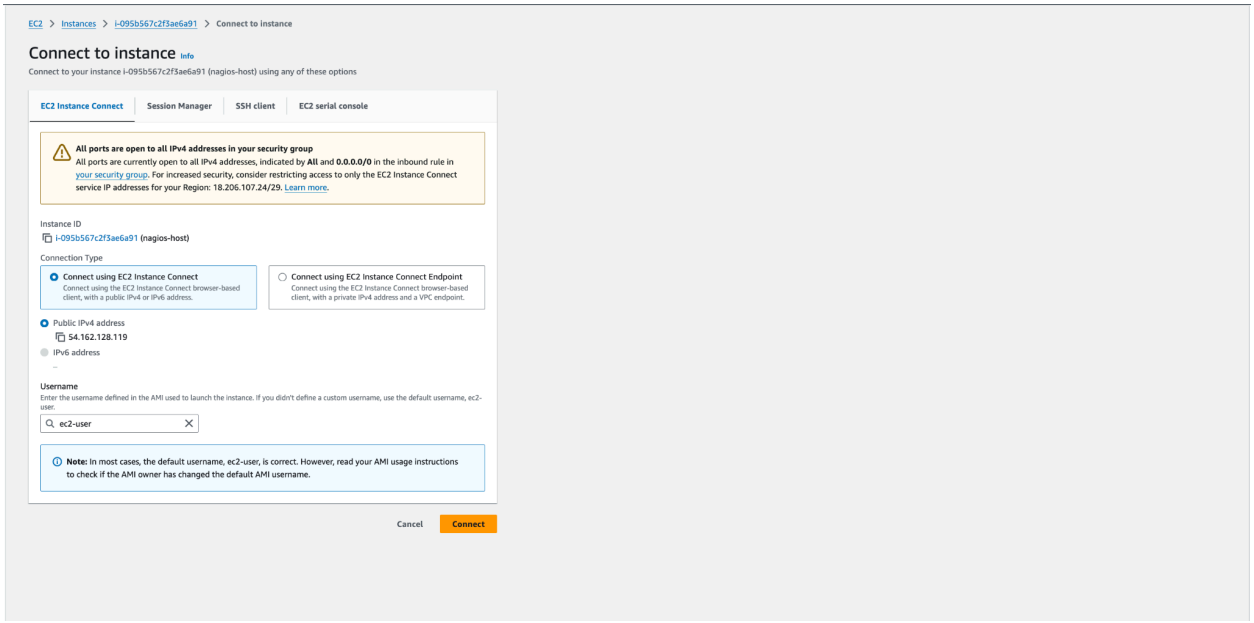


2. Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere.



You have to edit the inbound rules of the specified Security Group for this.

3. SSH into Your EC2 instance or simply use EC2 Instance Connect from the browser.




```
[ec2-user@ip-172-31-34-108 ~]$ sudo yum update
Last metadata expiration check: 0:04:33 ago on Fri Oct  4 04:35:02 2024.
Dependencies resolved.
Nothing to do.
Complete!
```

sudo yum update

```
[ec2-user@ip-172-31-91-91 ~]$
sudo yum update
Last metadata expiration check: 0:19:03 ago on Sun Sep 29 06:56:15 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-91-91 ~]$
```

sudo yum install httpd php

```
[ec2-user@ip-172-31-91-91 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:19:29 ago on Sun Sep 29 06:56:15 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.amzn2023.0.1                   amazonlinux         10 k
Installing dependencies:
 apr                        x86_64             1.7.2-2.amzn2023.0.2                    amazonlinux        129 k
 apr-util                   x86_64             1.6.3-1.amzn2023.0.1                    amazonlinux         98 k
 generic-logos-httpd        noarch             18.0.0-12.amzn2023.0.3                  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         14 k
 httpd-tools                x86_64             2.4.62-1.amzn2023                       amazonlinux         81 k
 libbrotli                  x86_64             1.0.9-4.amzn2023.0.2                    amazonlinux        315 k
 libsodium                  x86_64             1.0.19-4.amzn2023                       amazonlinux        176 k
 libxslt                    x86_64             1.1.34-5.amzn2023.0.2                   amazonlinux        241 k
 mailcap                    noarch             2.1.49-3.amzn2023.0.3                   amazonlinux         33 k
 nginx-filesystem           noarch             1:1.24.0-1.amzn2023.0.4                 amazonlinux        9.8 k
 php8.3-cli                 x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        3.7 M
 php8.3-common              x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        737 k
 php8.3-process             x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux         45 k
 php8.3-xml                 x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        154 k
Installing weak dependencies:
 apr-util-openssl           x86_64             1.6.3-1.amzn2023.0.1                    amazonlinux         17 k
 mod_http2                  x86_64             2.0.27-1.amzn2023.0.3                   amazonlinux        166 k
 mod_lua                    x86_64             2.4.62-1.amzn2023                       amazonlinux         61 k
 php8.3-fpm                 x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        1.9 M
 php8.3-mbstring            x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        528 k
 php8.3-opcache             x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux        379 k
 php8.3-pdo                 x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux         89 k
 php8.3-sodium              x86_64             8.3.10-1.amzn2023.0.1                   amazonlinux         41 k

Transaction Summary
```

```
Total                                                                                22 MB/s |  10 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                            1/1
  Installing       : php8.3-common-8.3.10-1.amzn2023.0.1.x86_64                                                 1/25
  Installing       : apr-1.7.2-2.amzn2023.0.2.x86_64                                                            2/25
  Installing       : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64                                               3/25
  Installing       : apr-util-1.6.3-1.amzn2023.0.1.x86_64                                                       4/25
  Installing       : mailcap-2.1.49-3.amzn2023.0.3.noarch                                                       5/25
  Running scriptlet: httpd-filesystem-2.4.62-1.amzn2023.noarch                                                  6/25
```

sudo yum install gcc glibc glibc-common

```
[ec2-user@ip-172-31-34-108 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:06:01 ago on Fri Oct  4 04:35:02 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             Arch    Version                       Repository    Size
================================================================================
Installing:
 gcc                 x86_64  11.4.1-2.amzn2023.0.2         amazonlinux   32 M
Installing dependencies:
 annobin-docs        noarch  10.93-1.amzn2023.0.1          amazonlinux   92 k
 annobin-plugin-gcc  x86_64  10.93-1.amzn2023.0.1          amazonlinux  887 k
 cpp                 x86_64  11.4.1-2.amzn2023.0.2         amazonlinux   10 M
 gc                  x86_64  8.0.4-5.amzn2023.0.2          amazonlinux  105 k
 glibc-devel         x86_64  2.34-52.amzn2023.0.11         amazonlinux   27 k
 glibc-headers-x86   noarch  2.34-52.amzn2023.0.11         amazonlinux  427 k
 guile22             x86_64  2.2.7-2.amzn2023.0.3          amazonlinux  6.4 M
 kernel-headers      x86_64  6.1.109-118.189.amzn2023      amazonlinux  1.4 M
```

```
Installed:
  annobin-docs-10.93-1.amzn2023.0.1.noarch
  annobin-plugin-gcc-10.93-1.amzn2023.0.1.x86_64
  cpp-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

Complete!
```

sudo yum install gd gd-devel

```
[ec2-user@ip-172-31-34-108 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:06:19 ago on Fri Oct  4 04:35:02 2024.
Dependencies resolved.
================================================================================
 Package                  Arch   Version                    Repository   Size
================================================================================
Installing:
 gd                       x86_64 2.3.3-5.amzn2023.0.3       amazonlinux 139 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 314 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 214 k
 cairo                    x86_64 1.17.6-2.amzn2023.0.1      amazonlinux 684 k
 cmake-filesystem         x86_64 3.22.2-1.amzn2023.0.4      amazonlinux  16 k
 fontconfig               x86_64 2.13.94-2.amzn2023.0.2     amazonlinux 273 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 2.74.7-689.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  21 k
 harfbuzz                 x86_64 7.0.0-2.amzn2023.0.1       amazonlinux 868 k
 harfbuzz-devel           x86_64 7.0.0-2.amzn2023.0.1       amazonlinux 404 k
 harfbuzz-icu             x86_64 7.0.0-2.amzn2023.0.1       amazonlinux  18 k
 jbigkit-libs             x86_64 2.1-21.amzn2023.0.2        amazonlinux  54 k
 langpacks-core-font-en   noarch 3.0-21.amzn2023.0.4        amazonlinux  10 k
 libICE                   x86_64 1.0.10-6.amzn2023.0.2      amazonlinux  71 k
 libSM                    x86_64 1.2.3-8.amzn2023.0.2       amazonlinux  42 k
 libX11                   x86_64 1.7.2-3.amzn2023.0.4       amazonlinux 657 k
```

```
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.amzn2023.0.1.x86_64
  freetype-devel-2.13.2-5.amzn2023.0.1.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.amzn2023.0.2.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                 libblkid-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!
```

5. Create a new Nagios User with its password. You'll have to enter the password twice for confirmation.

sudo adduser -m nagios
sudo passwd nagios
(password : *hello123)*

```
[ec2-user@ip-172-31-34-108 ~]$ sudo adduser -m nagios
sudo passwd nagios
Changing password for user nagios.
New password:
[ec2-user@ip-172-31-34-108 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dic
tionary word
Retype new password:
passwd: all authentication tokens updated successfully.
```

6. Create a new user group
sudo groupadd nagcmd

```
[ec2-user@ip-172-31-34-108 ~]$ 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

```
[ec2-user@ip-172-31-34-108 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
```

8. Create a new directory for Nagios downloads
mkdir ~/downloads
cd ~/downloads

```
[ec2-user@ip-172-31-34-108 ~]$ mkdir ~/downloads
cd ~/downloads
```

9. 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-172-31-34-108 downloads]$ wget https://assets.nagios.com/downlo
ads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-10-04 04:42:39--  https://assets.nagios.com/downloads/nagioscore/rele
ases/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... con
nected.
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.g 100%[===============>]   1.97M  5.31MB/s    in 0.4s

2024-10-04 04:42:39 (5.31 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/20654
73]
```

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

```
[ec2-user@ip-172-31-34-108 downloads]$ wget https://nagios-plugins.org/downl
oad/nagios-plugins-2.4.11.tar.gz
--2024-10-04 04:42:47--  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-2.4 100%[=================>]   2.62M  5.79MB/s    in 0.5s

2024-10-04 04:42:48 (5.79 MB/s) - ‘nagios-plugins-2.4.11.tar.gz’ saved [2753
049/2753049]

[ec2-user@ip-172-31-34-108 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
```

10. Use tar to unzip and change to that directory.
tar zxvf nagios-4.5.5.tar.gz

```
[ec2-user@ip-172-31-91-91 downloads]$ tar zxvf nagios-4.0.8.tar.gz
nagios-4.0.8/
nagios-4.0.8/.gitignore
nagios-4.0.8/Changelog
nagios-4.0.8/INSTALLING
nagios-4.0.8/LEGAL
nagios-4.0.8/LICENSE
nagios-4.0.8/Makefile.in
nagios-4.0.8/README
nagios-4.0.8/README.asciidoc
nagios-4.0.8/THANKS
nagios-4.0.8/UPGRADING
nagios-4.0.8/base/
nagios-4.0.8/base/.gitignore
```

11. Run the configuration script with the same group name you previously created.
./configure --with-command-group=nagcmd

Here we get an error

```
[ec2-user@ip-172-31-34-108 downloads]$ ./configure --with-command-group=nagc
md
-bash: ./configure: No such file or directory
```

Solution

Navigate to nagios folder in downloads
cd nagios-4.5.5

```
[ec2-user@ip-172-31-34-108 downloads]$ cd nagios-4.5.5
```

Error 2: Cannot find SSL headers.
Solution: Install openssl dev library

Steps:
sudo yum install openssl-devel

```
[ec2-user@ip-172-31-91-91 nagios-4.5.5]$ sudo yum install openssl-devel
Last metadata expiration check: 2:24:05 ago on Sun Sep 29 06:56:15 2024.
Dependencies resolved.
================================================================================
 Package          Arch      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

Total download size: 3.0 M
Installed size: 4.7 M
Is this ok [y/N]: y
Downloading Packages:
```

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

```
[ec2-user@ip-172-31-34-108 nagios-4.5.5]$ ./configure --with-command-group=n
agcmd
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
```

12. Compile the source code.
make all

```
[ec2-user@ip-172-31-34-108 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’,
    inlined from ‘get_worker’ at workers.c:277:12:
workers.c:253:17: warning: ‘%s’ directive argument is null [-Wformat-overflo
w=]
  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
```

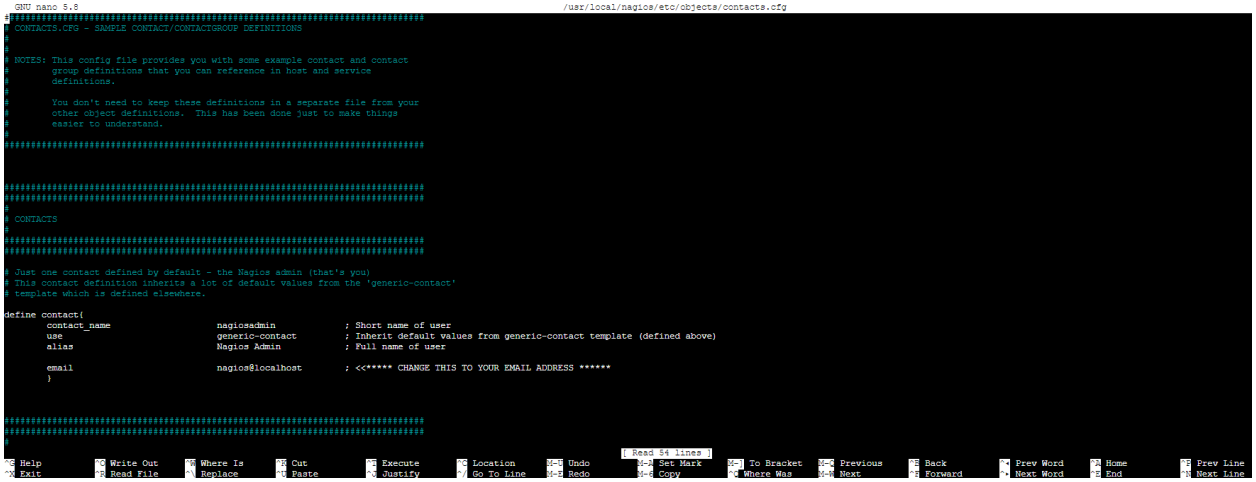
13. Install binaries, init script and sample config files. Lastly, set permissions on the external command directory.
sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode

```
[ec2-user@ip-172-31-34-108 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/b
in
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiostats /usr/local/nagi
os/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/n
agios/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/me
dia
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/st
ylesheets
```

14. Edit the config file and change the email address.

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

```
[ec2-user@ip-172-31-34-108 nagios-4.5.5]$ sudo nano /usr/local/nagios/etc/ob
jects/contacts.cfg
```



In this page, change email address:

```
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.brijesh.sharma@ves.ac.in  ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS ******
        }
```

15. Configure the web interface.

sudo make install-webconf

```
[ec2-user@ip-172-31-34-108 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/nagio
s.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

```
[ec2-user@ip-172-31-34-108 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
```

Password: *hello123*

17. Restart Apache

sudo service httpd restart

```
[ec2-user@ip-172-31-34-108 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

```
[ec2-user@ip-172-31-34-108 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
```

19. Compile and install plugins

cd nagios-plugins-2.4.11

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

```
[ec2-user@ip-172-31-34-108 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... yes
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 gcc understands -c and -o together... yes
checking whether make supports the include directive... yes (GNU style)
```

make
sudo make install

```
[ec2-user@ip-172-31-34-108 nagios-plugins-2.4.11]$ make
sudo make install
make  all-recursive
make[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
Making all in gl
make[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/
gl'
rm -f alloca.h-t alloca.h && \
{ echo '/* DO NOT EDIT! GENERATED AUTOMATICALLY! */'; \
  cat ./alloca.in.h; \
} > alloca.h-t && \
mv -f alloca.h-t alloca.h
rm -f c++defs.h-t c++defs.h && \
sed -n -e '/_GL_CXXDEFS/,$p' \
  < ../build-aux/snippet/c++defs.h \
  > c++defs.h-t && \
mv c++defs.h-t c++defs.h
rm -f warn-on-use.h-t warn-on-use.h && \
sed -n -e '/^.ifndef/,$p' \
```

```
fi
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/p
o'
make[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
make[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
```

20. Start Nagios

Add Nagios to the list of system services

sudo chkconfig --add nagios

sudo chkconfig nagios on

```
[ec2-user@ip-172-31-34-108 nagios-plugins-2.4.11]$ sudo chkconfig --add nagi
os
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.
```

Verify the sample configuration files

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

```
[ec2-user@ip-172-31-34-108 nagios-plugins-2.4.11]$ sudo /usr/local/nagios/bin/nagios -v /us
r/local/nagios/etc/nagios.cfg

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
```

sudo service nagios start

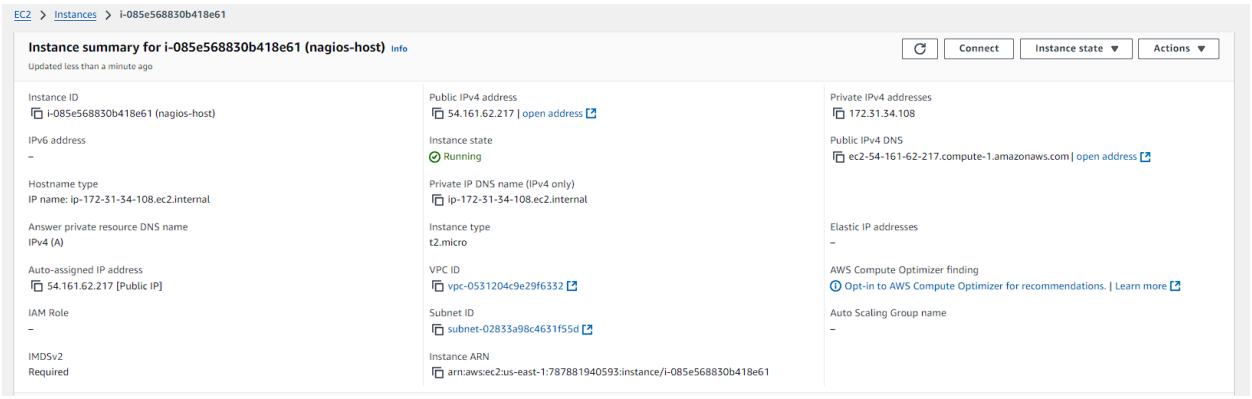
```
[ec2-user@ip-172-31-34-108 nagios-plugins-2.4.11]$ sudo service nagios startRedirecting to
/bin/systemctl start nagios.service
```

21. Check the status of Nagios

sudo systemctl status nagios

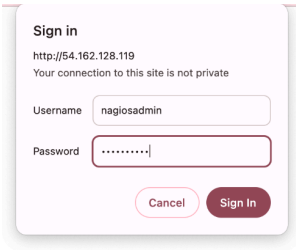
```
[ec2-user@ip-172-31-34-108 nagios-plugins-2.4.11]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; prese>     Active: active (runni
ng) since Fri 2024-10-04 04:48:01 UTC; 6s ago
       Docs: https://www.nagios.org/documentation
    Process: 63465 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/>    Process: 63466 ExecSta
rt=/usr/local/nagios/bin/nagios -d /usr/local/nag>   Main PID: 63467 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 79ms
     CGroup: /system.slice/nagios.service
             ├─63467 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/>             ├─63468 /usr/
local/nagios/bin/nagios --worker /usr/local/nagio>             ├─63469 /usr/local/nagios/bin/nagios --
worker /usr/local/nagio>             ├─63470 /usr/local/nagios/bin/nagios --worker /usr/local/nagio>
           ├─63471 /usr/local/nagios/bin/nagios --worker /usr/local/nagio>             └─63472 /usr/lo
cal/nagios/bin/nagios -d /usr/local/nagios/etc/>
Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: qh: Socket '/u>Oct 04 04:48:01 ip-172-31-
34-108.ec2.internal nagios[63467]: qh: core query>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios
[63467]: qh: echo servi>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: qh: help for t>Oc
t 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Success>Oct 04 04:48:01 ip-172-31-34
-108.ec2.internal nagios[63467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[6
3467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Registr>Oct
04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-1
08.ec2.internal nagios[63467]: Successfully l>lines 1-28/28 (END)
[ec2-user@ip-172-31-34-108 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; prese>     Active: active (runni
ng) since Fri 2024-10-04 04:48:01 UTC; 30s ago
       Docs: https://www.nagios.org/documentation
    Process: 63465 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/>    Process: 63466 ExecSta
rt=/usr/local/nagios/bin/nagios -d /usr/local/nag>   Main PID: 63467 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 82ms
     CGroup: /system.slice/nagios.service
             ├─63467 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/>             ├─63468 /usr/
local/nagios/bin/nagios --worker /usr/local/nagio>             ├─63469 /usr/local/nagios/bin/nagios --
worker /usr/local/nagio>             ├─63470 /usr/local/nagios/bin/nagios --worker /usr/local/nagio>
           ├─63471 /usr/local/nagios/bin/nagios --worker /usr/local/nagio>             └─63472 /usr/lo
cal/nagios/bin/nagios -d /usr/local/nagios/etc/>
Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: qh: Socket '/u>Oct 04 04:48:01 ip-172-31-
34-108.ec2.internal nagios[63467]: qh: core query>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios
[63467]: qh: echo servi>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: qh: help for t>Oc
t 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Success>Oct 04 04:48:01 ip-172-31-34
-108.ec2.internal nagios[63467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[6
3467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Registr>Oct
04 04:48:01 ip-172-31-34-108.ec2.internal nagios[63467]: wproc: Registr>Oct 04 04:48:01 ip-172-31-34-1
08.ec2.internal nagios[63467]: Successfully l>lines 1-28/28 (END)client_loop: send disconnect: Connect
ion reset
```

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

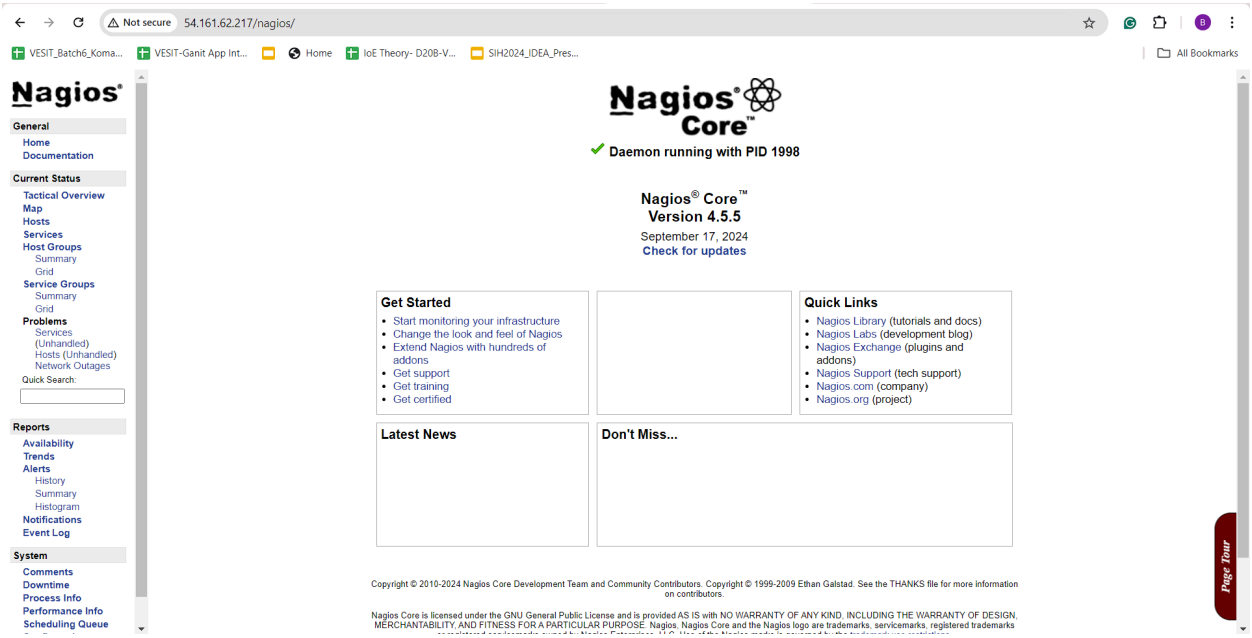


23. Open up your browser and look for http://<your_public_ip_address>/nagios

Enter username as nagiosadmin and password which you set in Step 16.



24. After entering the correct credentials, you will see this page.



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

Conclusion:

In this practical, we successfully installed and configured Nagios Core along with Nagios plugins and NRPE on an Amazon EC2 instance. We created a Nagios user, set up necessary permissions, and resolved common installation errors. Finally, we verified the setup by accessing the Nagios web interface, confirming that our monitoring system was fully operational.