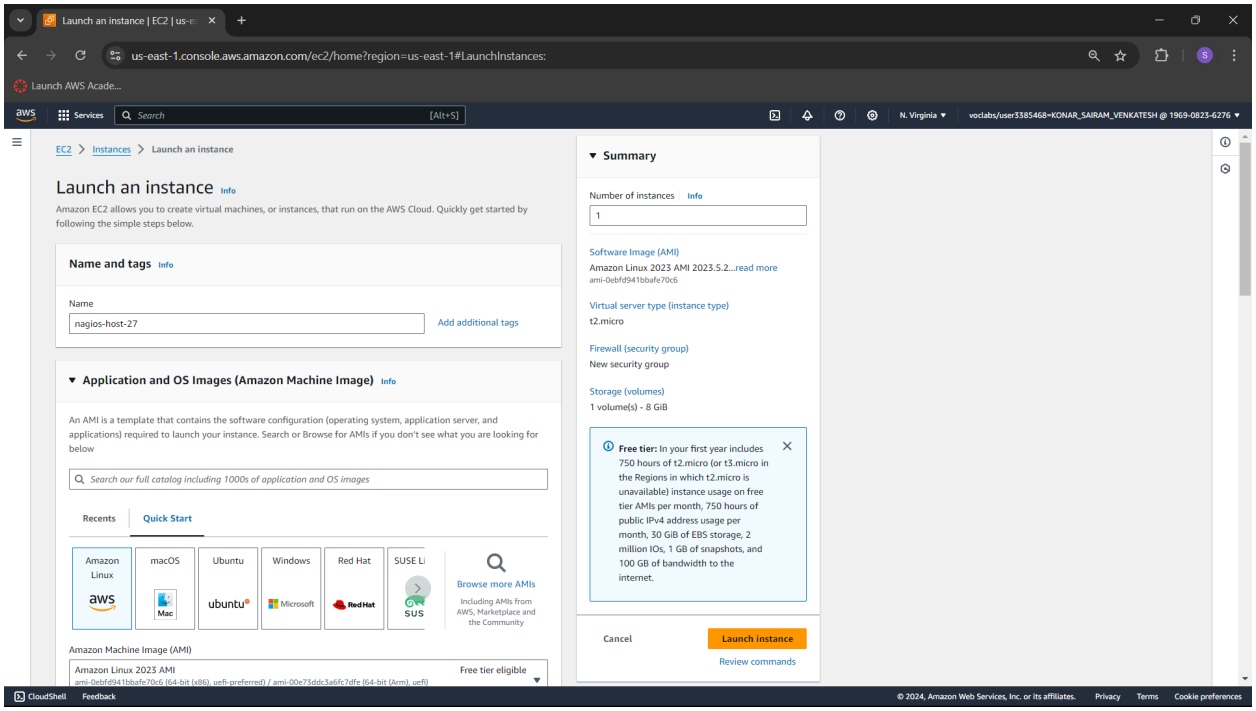
Sairam Konar D15C 27 2024-25

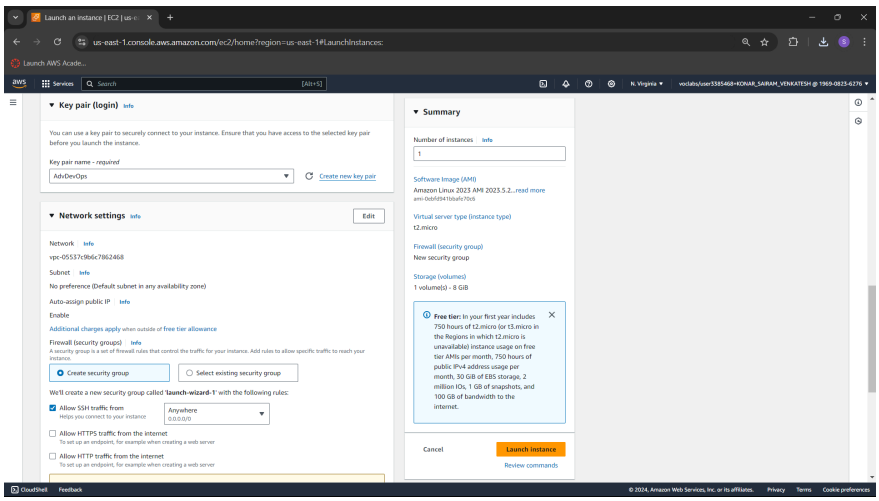
Aim: To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Step 1: Login to your AWS account. Search for EC2 on services. Open the interface and click on Create Instance.



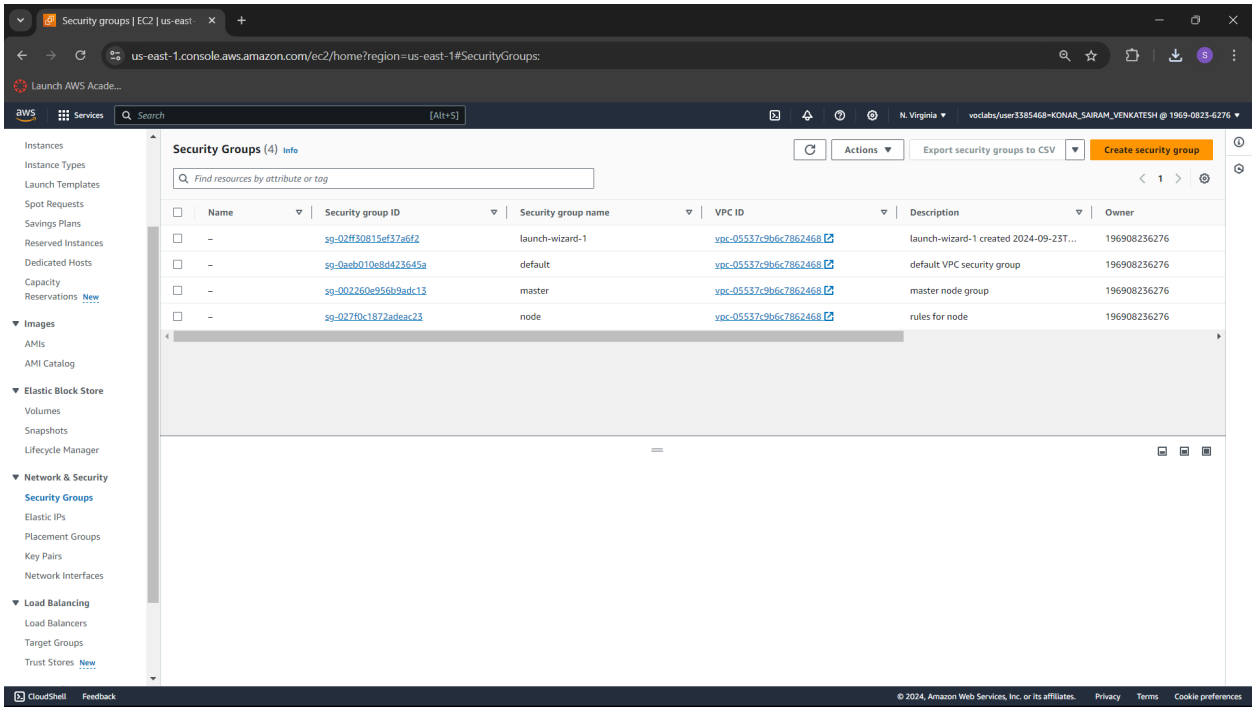
Select The OS Image as Amazon Linux.

Step 2: If you do not have a private key created or a .pem file created, click on create a key pair. Else select the key pair that you had created before. (Make sure you know where the .pem file for that key is present on your system)

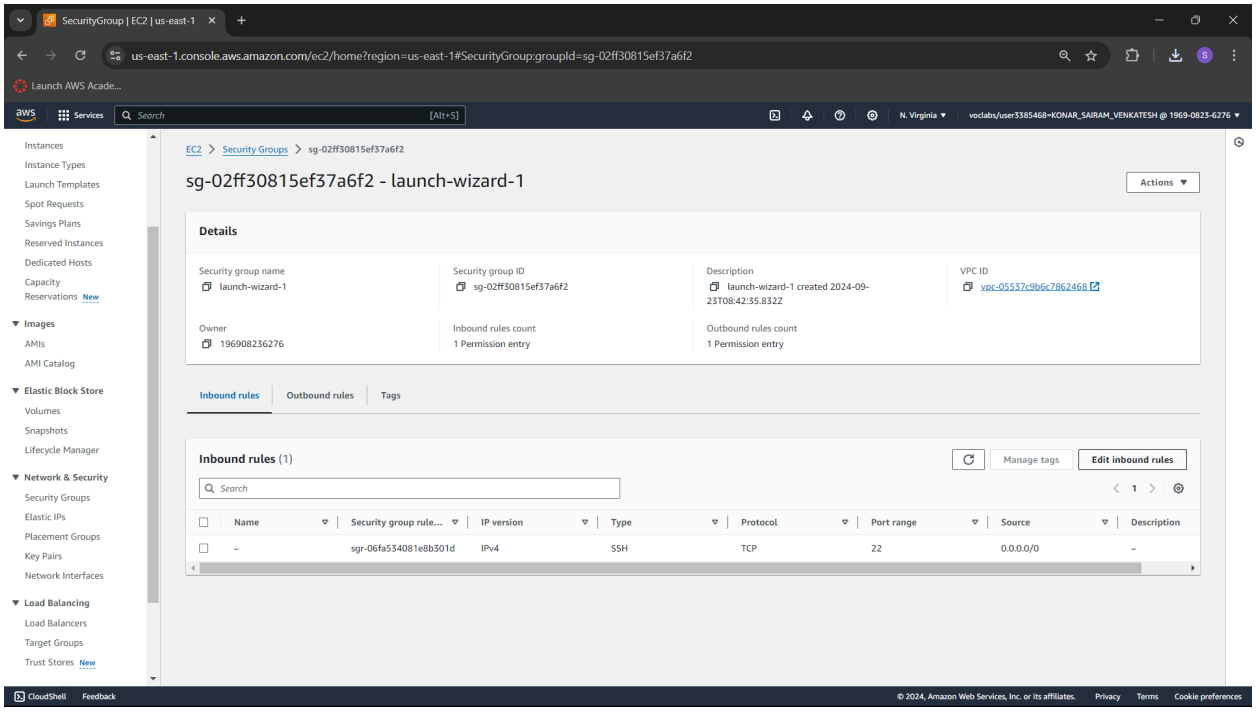


AWS will create a security group for this instance. Keep the name of that instance saved.

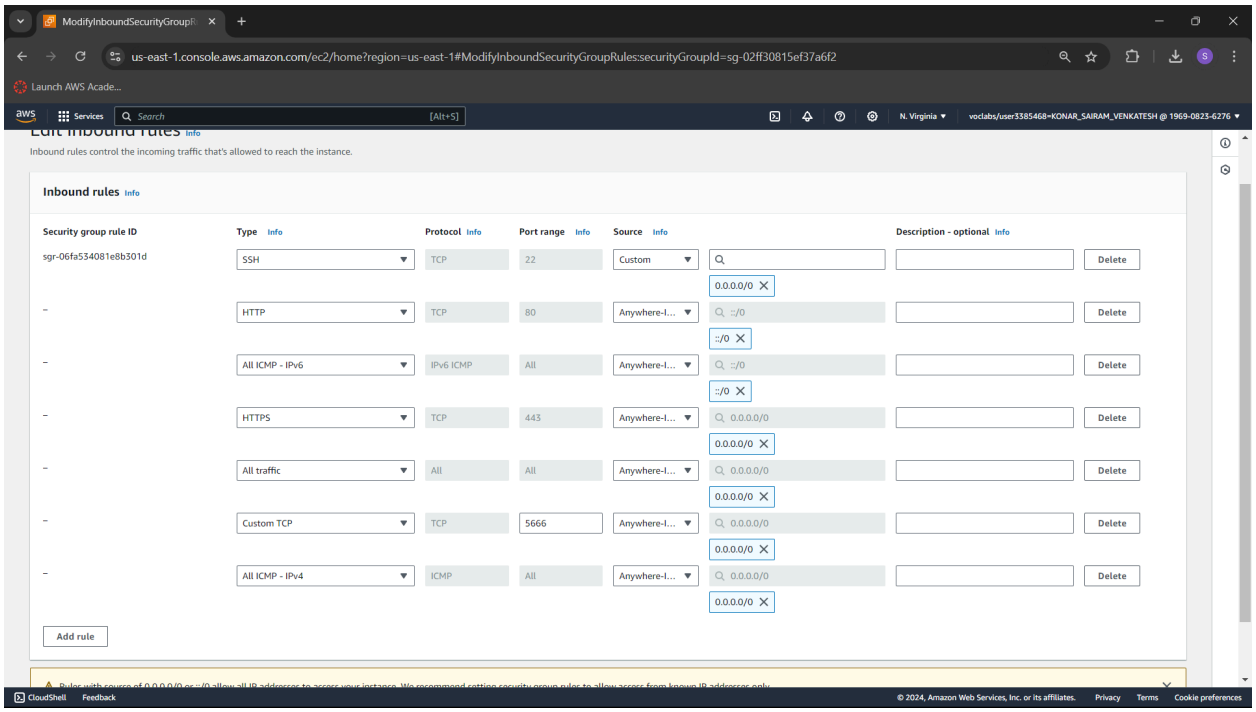
Step 3: After creating the instance, click on Security Groups from the left side pane. Find the security group that was created for your instance. Click on the instance ID for that group.



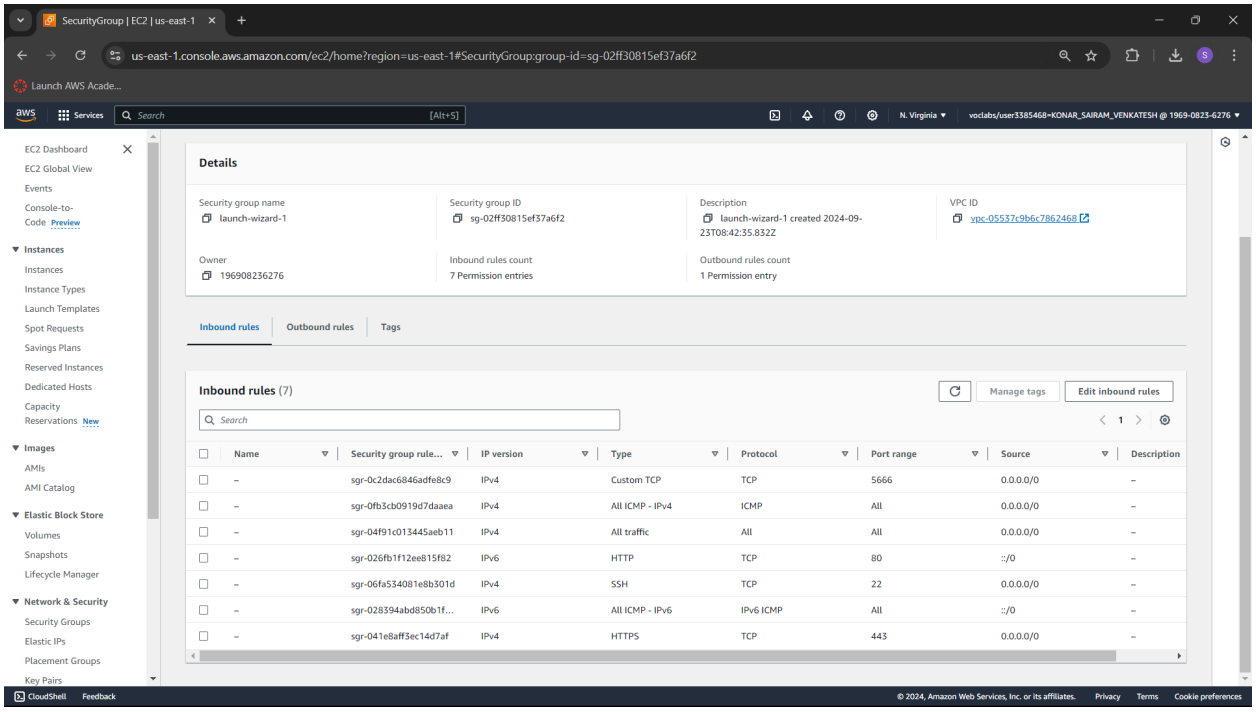
Here, click on Edit Inbound Rules.



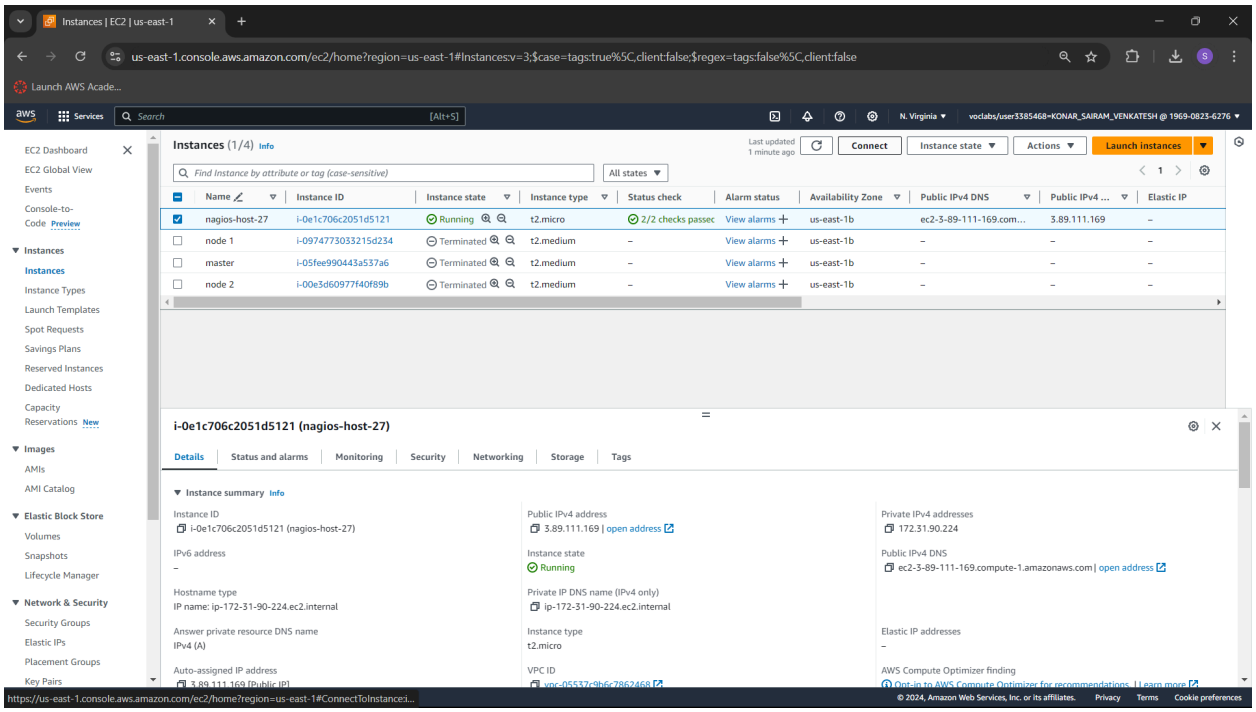
Now, click on add rules, and add teh rules for the following protocols:
HTTP, All ICMP - IPv6, HTTPS, All traffic, Custom TCP (Port 5666), All ICMP - IPv4



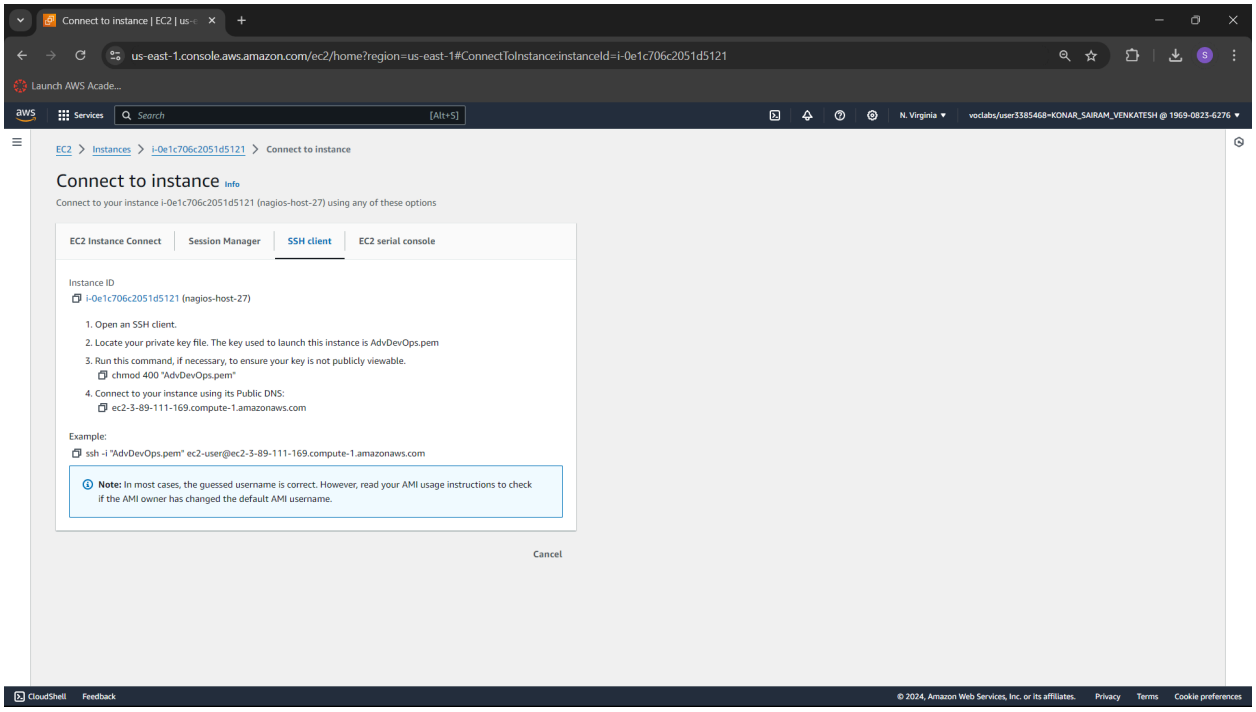
Click on save. This will add all the inbound rules to the security group.



Step 4: Now come back to the instances screen. Click on the instance ID of your instance. Then click on Connect.

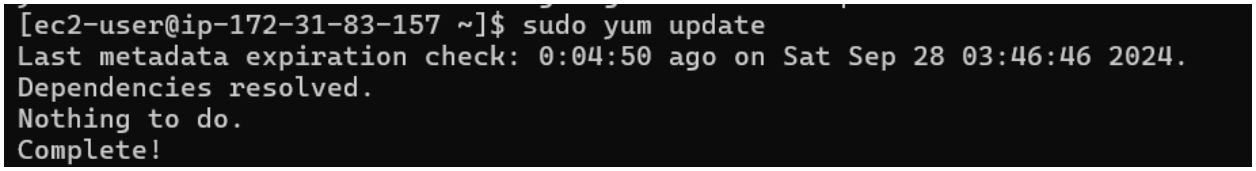


Click on SSH client. Copy the example command.

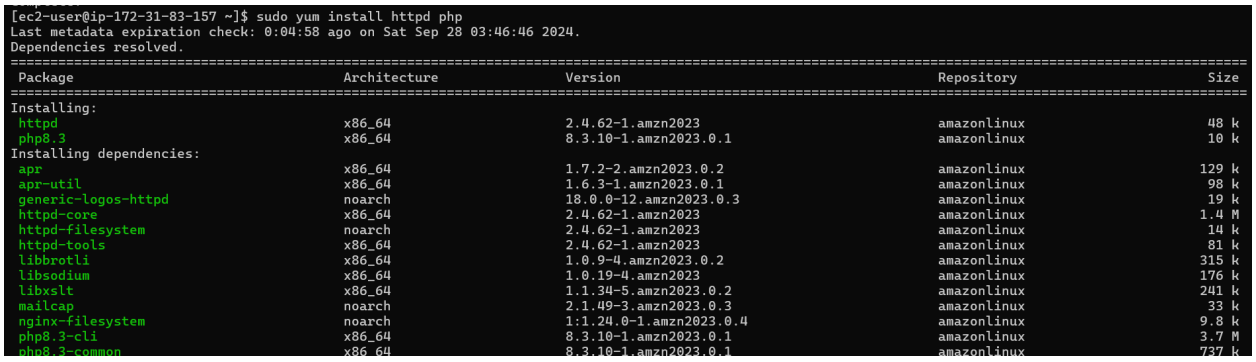


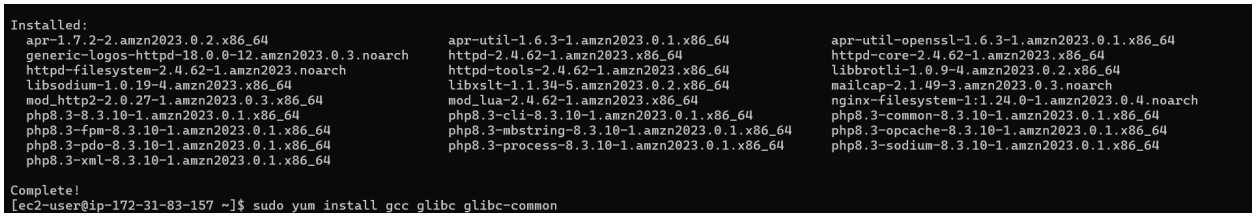
Step 5: Now, we have to connect our local OS terminal to the instance using SSH. For this,
Open terminal wher the private key file is located (.pem)
Paste the copied SSH command and run it.

Step 6: Now we start working on this terminal. First run the command
sudo yum update
This command will check for any updates for the YUM library.



Step 7: We are going to install an Apache server and a PHP on this instance. For that, run this command.
sudo yum install httpd php





Step 8: Next we install C/C++ compiler (GCC) along with the necessary C libraries required for compiling and running C programs. Use the following command.
sudo yum install gcc glibc glibc-common

```
[ec2-user@ip-172-31-83-157 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:05:58 ago on Sat Sep 28 03:46:46 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.amzn2023.0.2    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
 libmpc                  x86_64          1.2.1-2.amzn2023.0.2     amazonlinux  62 k
 libtool-ltdl            x86_64          2.4.7-1.amzn2023.0.3     amazonlinux  38 k
 libxcrypt-devel         x86_64          4.4.33-7.amzn2023        amazonlinux  32 k

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!
[ec2-user@ip-172-31-83-157 ~]$
```

Step 9: We would also need GD library and its development tools. For that, run this command
sudo yum install gd gd-devel

```
[ec2-user@ip-172-31-83-157 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:11:36 ago on Sat Sep 28 03:46:46 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture    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

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-devel-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!
[ec2-user@ip-172-31-83-157 ~]$
```

Step 10: Now, we create a user called as ‘nagios’ and make sure that it has a home directory, and set up a password for it.
sudo adduser -m nagios
sudo passwd nagios

```
[ec2-user@ip-172-31-83-157 ~]$ sudo adduser -m nagios
sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
```

Step 11: Create a user group called as ‘nagcmd’ to execute nagios commands.
sudo groupadd nagcmd

```
[ec2-user@ip-172-31-83-157 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-83-157 ~]$
```

Step 12: Add users apache and nagios to this user group.
sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache

```
[ec2-user@ip-172-31-83-157 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-83-157 ~]$
```

Step 13: We create a directory downloads, to store the files of nagios server that are downloaded.
mkdir ~/downloads
cd ~/downloads

```
[ec2-user@ip-172-31-83-157 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-83-157 downloads]$
```

Step 14:Now we need to install the latest versions of nogios-core and nagios-plugins. Go to the respective websites and check whether a better version is available.If newer versions are available, then right click on the download button → Copy link address.
Paste this link address in place of the current link in command.
If not run these commands.

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

```
[ec2-user@ip-172-31-83-157 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-09-28 04:04:23--  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%[=============================================================================>]   1.97M  5.36MB/s    in 0.4s

2024-09-28 04:04:24 (5.36 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/2065473]

[ec2-user@ip-172-31-83-157 downloads]$
```

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

```
[ec2-user@ip-172-31-83-157 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-09-28 04:06:15--  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  5.22MB/s    in 0.5s

2024-09-28 04:06:16 (5.22 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]

[ec2-user@ip-172-31-83-157 downloads]$
```

Step 15:Now, we need to extract nagios-core file into the same directory. For that, we will use tar command.
tar zxvf nagios-4.5.5.tar.gz

```
[ec2-user@ip-172-31-83-157 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
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/xdata/.gitignore
nagios-4.5.5/xdata/Makefile.in
nagios-4.5.5/xdata/xcddefault.c
nagios-4.5.5/xdata/xcddefault.h
nagios-4.5.5/xdata/xodtemplate.c
nagios-4.5.5/xdata/xodtemplate.h
nagios-4.5.5/xdata/xpddefault.c
nagios-4.5.5/xdata/xpddefault.h
nagios-4.5.5/xdata/xrddefault.c
nagios-4.5.5/xdata/xrddefault.h
nagios-4.5.5/xdata/xsddefault.c
nagios-4.5.5/xdata/xsddefault.h
[ec2-user@ip-172-31-83-157 downloads]$
```

Step 16: We need to ensure that Nagios uses a specific group (in this case, `nagcmd`) for executing external commands.
./configure --with-command-group=nagcmd

An error could be encountered here: ./configure: no such path or directory

Solution: Navigate to the nagios-4.5.5 folder in downloads. (version could vary)

Steps: ls

```
[ec2-user@ip-172-31-83-157 downloads]$ ls
nagios-4.5.5  nagios-4.5.5.tar.gz  nagios-plugins-2.4.11.tar.gz
```

- cd nagios-4.5.5 (use the version shown by your ls command)
- ./configure --with-command-group=nagcmd

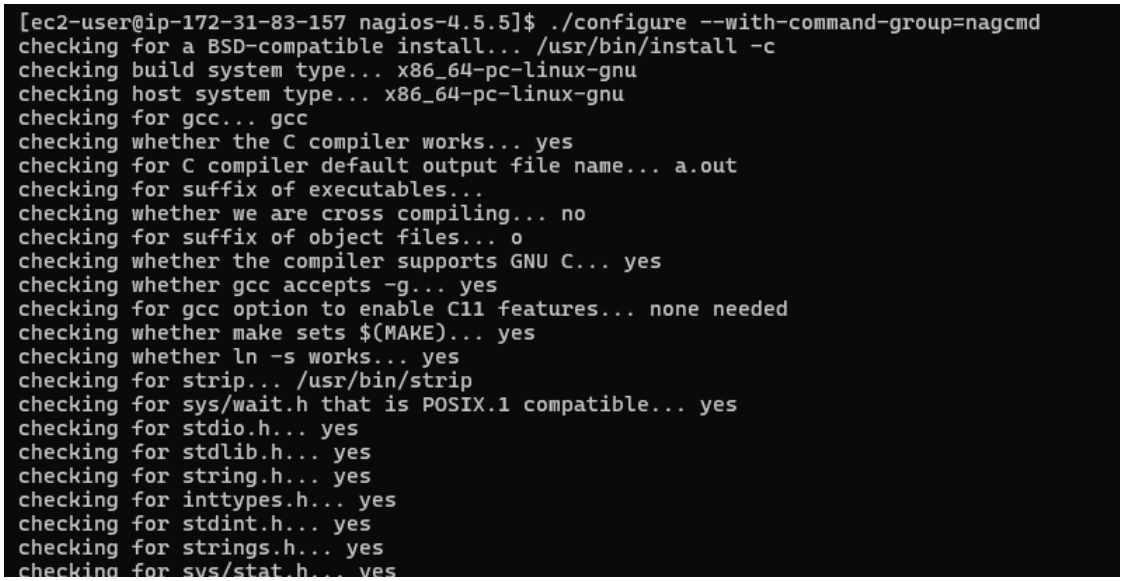
Another error could be Cannot find SSL headers.

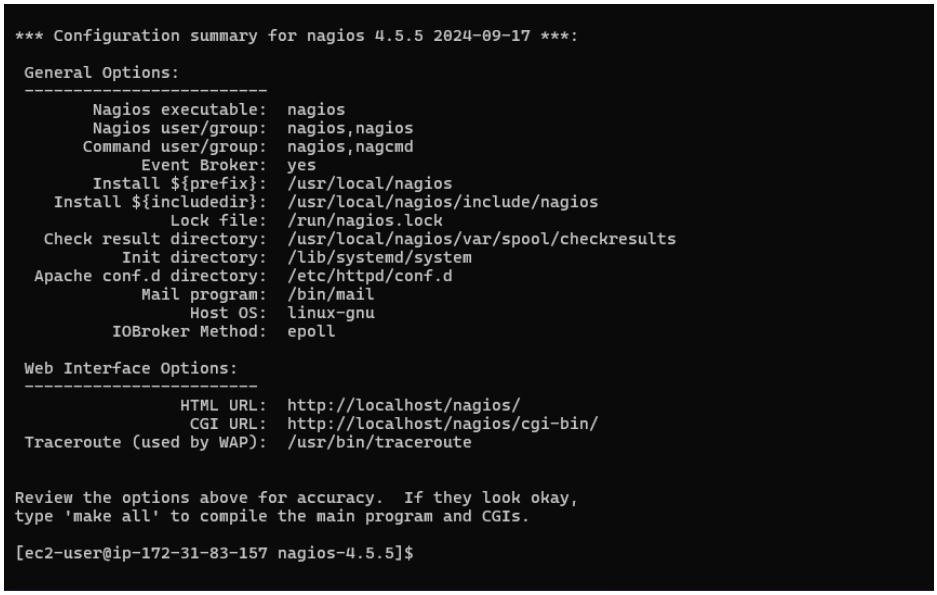
To solve this, we need to install OpenSSL Dev Library

Steps:

sudo yum install openssl-devel

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





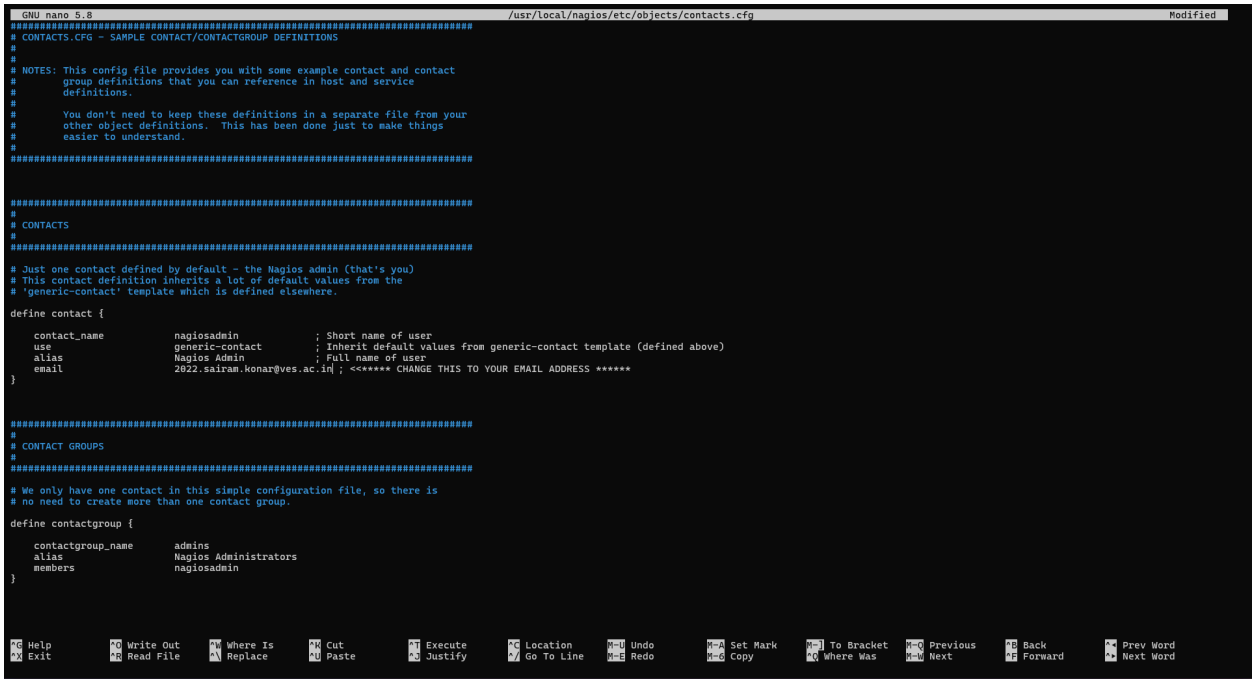
Step 17: We need to compile all components of this software according to the instruction in the Makefile. For that, use this command:
make all

Then,
sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode

```
[ec2-user@ip-172-31-83-157 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 nagiostats /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
```

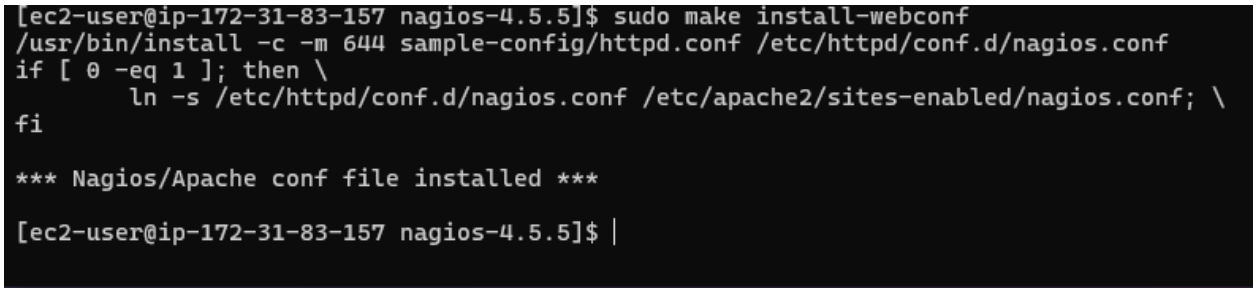
```
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5'
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

Remember, these are *SAMPLE* config files.  You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.

/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 ***
```

Step 18: We need to update the email linked with this server to our email for it to send notifications (if any needed).
sudo nano /usr/local/nagios/etc/objects/contacts.cfg



Here, change the email under ‘define contact{}’ to your email address.
To save this use the following shortcut sequence CTRL+O→Enter→CTRL+X.
CTRL+O: Overwrite the existing file with edited file
CTRL+X: Exit nano editor.

Step 19: We need to install the necessary configuration files for the Nagios web interface. Run the following command.
sudo make install-webconf



Step 20: Now we need to setup a user to access this nagios web interface. So we run this command to create a user called ‘nagiosadmin’.

Keep this username and password saved as it is needed to login to the web interface.
sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin

```
[ec2-user@ip-172-31-83-157 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
```

Step 21: Restart the apache server to apply all the recent configurations.
sudo service httpd restart

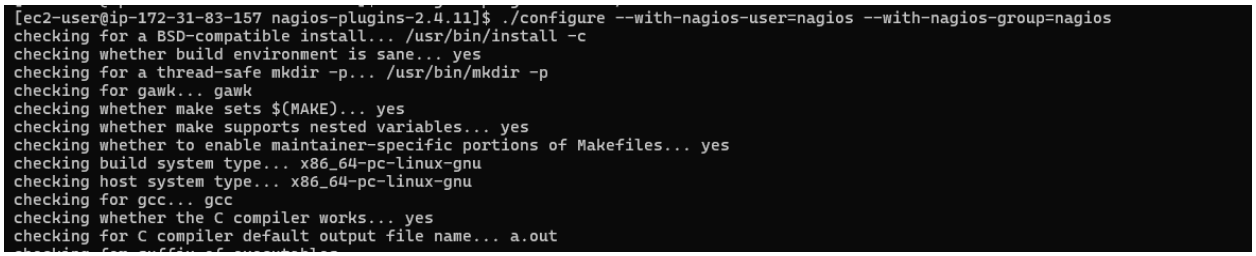
```
[ec2-user@ip-172-31-83-157 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
```

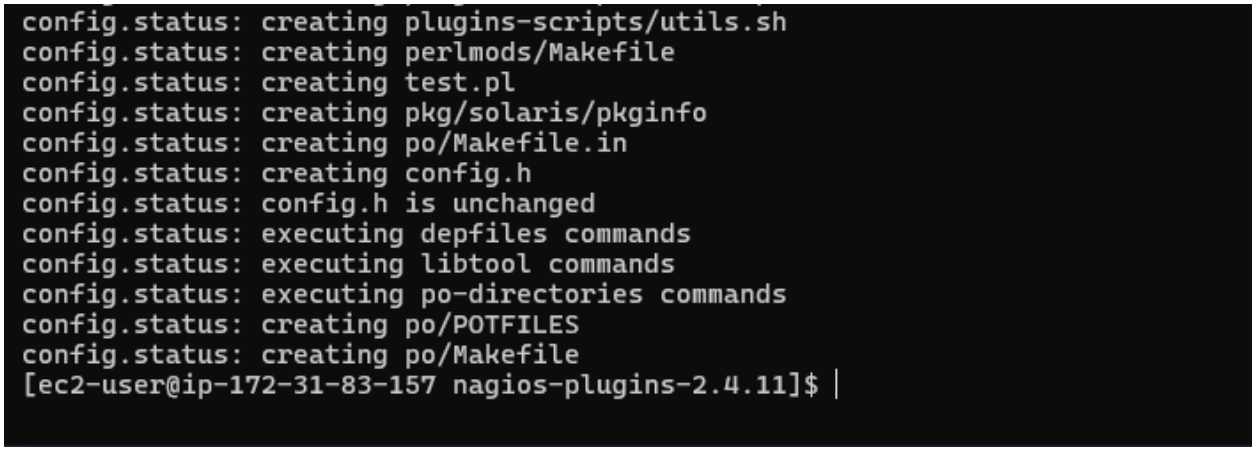
Step 22: Now we go back to the downloads folder and extract the files of nagios plugin.
cd ~/downloads
tar zxvf nagios-plugins-2.4.11.tar.gz (Version may vary)

```
[ec2-user@ip-172-31-83-157 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/
nagios-plugins-2.4.11/gl/m4/
```

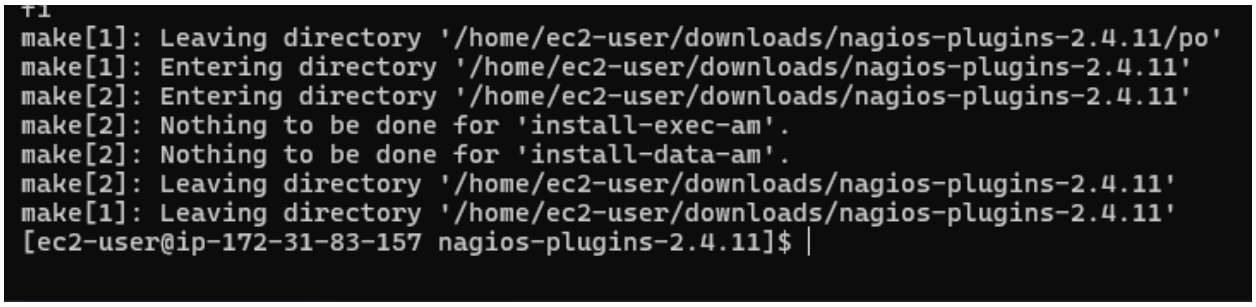
```
nagios-plugins-2.4.11/po/fr.gmo
nagios-plugins-2.4.11/po/de.gmo
nagios-plugins-2.4.11/po/nagios-plugins.pot
nagios-plugins-2.4.11/po/stamp-po
nagios-plugins-2.4.11/po/ChangeLog
nagios-plugins-2.4.11/po/LINGUAS
nagios-plugins-2.4.11/release
[ec2-user@ip-172-31-83-157 downloads]$
```

Step 23: Again, we need to install the configurations for these files.
cd nagios-plugins-2.4.11 (version may vary)
./configure --with-nagios-user=nagios --with-nagios-group=nagios





Step 24: We need to compile all components of this software according to the instruction in the Makefile. For that, use the commands:
make
sudo make install



Step 25: We need to register the Nagios service with the system, which would make it able to manage the server status. So run the following commands
sudo chkconfig --add nagios
sudo chkconfig nagios on

Step 26: We need to verify the Nagios configuration for any syntax errors or issues before starting or restarting the Nagios service.
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```
[ec2-user@ip-172-31-90-224 nagios-plugins-2.0.3]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.0.8
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-12-2014
License: GPL

Website: http://www.nagios.org
Reading configuration data...
Error in configuration file '/usr/local/nagios/etc/nagios.cfg' - Line 452 (Check result path '/usr/local/nagios/var/spool/checkresults' is not a valid directory)
   Error processing main config file!

[ec2-user@ip-172-31-90-224 nagios-plugins-2.0.3]$ sudo mkdir -p /usr/local/nagios/var/spool/checkresults
[ec2-user@ip-172-31-90-224 nagios-plugins-2.0.3]$ sudo chown nagios:nagios /usr/local/nagios/var/spool/checkresults
sudo chmod 775 /usr/local/nagios/var/spool/checkresults
[ec2-user@ip-172-31-90-224 nagios-plugins-2.0.3]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.0.8
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-12-2014
License: GPL
```

Error: Error in configuration file '/usr/local/nagios/etc/nagios.cfg' - Line 452 (Check result path '/usr/local/nagios/var/spool/checkresults' is not a valid directory)
It is an error in processing main config file!

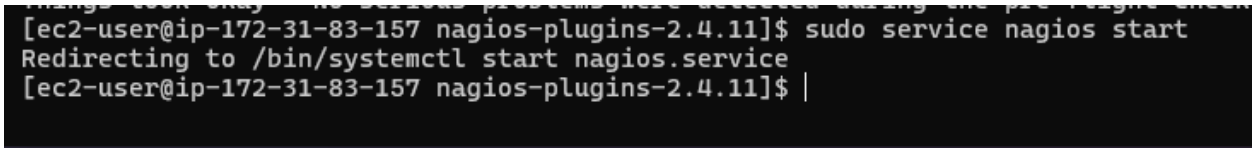
Solution: Create the missing directory, set the permissions, verify it.

- sudo mkdir -p /usr/local/nagios/var/spool/checkresults (Create)
- sudo chown nagios:nagios /usr/local/nagios/var/spool/checkresults
- sudo chmod 775 /usr/local/nagios/var/spool/checkresults (permissions)

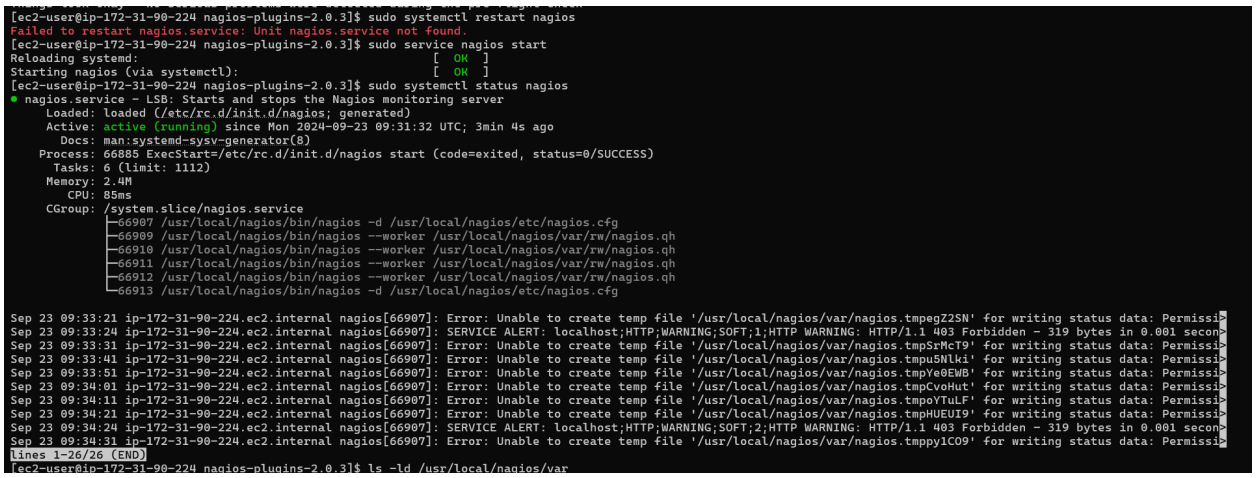
Now rerun the commmad
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.
[ec2-user@ip-172-31-83-157 nagios-plugins-2.4.11]$ sudo /usr/local/nagios/bin/nagios -v /usr/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

Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-83-157 nagios-plugins-2.4.11]$
```

Step 27:
sudo service nagios start



sudo systemctl status nagios



Error:Sep 23 09:34:31 ip-172-31-90-224.ec2.internal nagios[66907]: Error: Unable to create temp file '/usr/local/nagios/var/nagios.tmppy1CO9' for writing status data: Permissi>

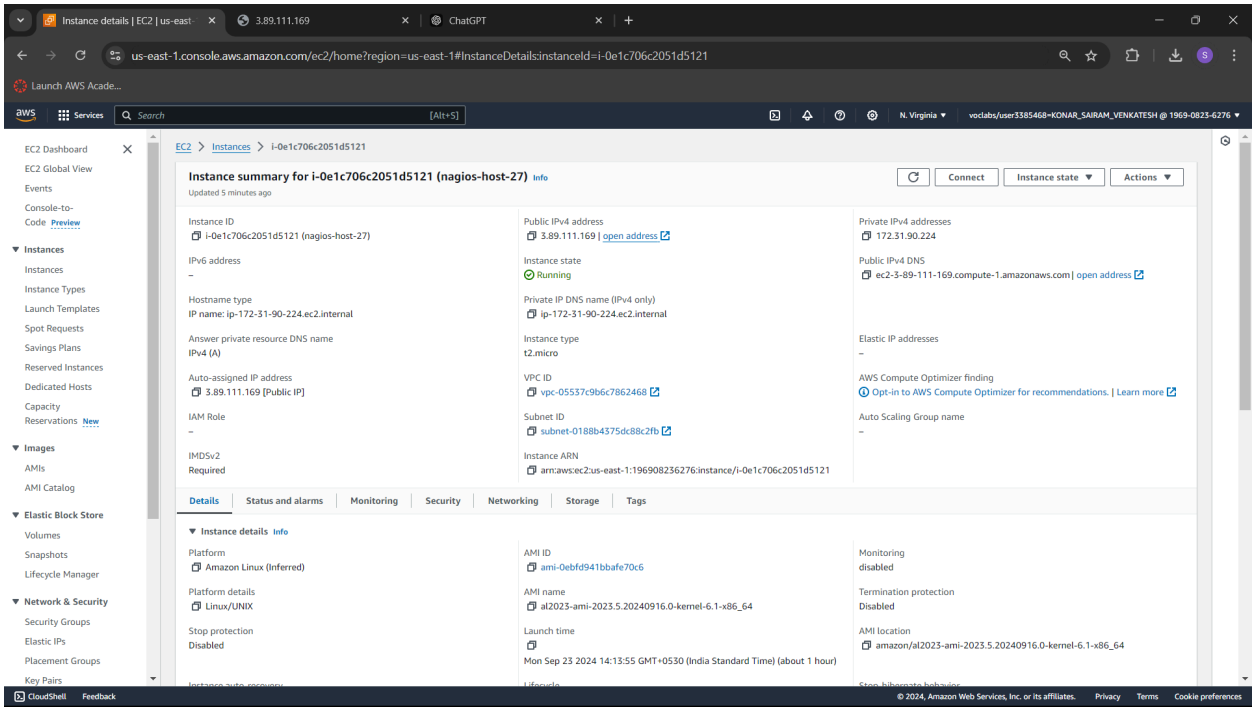
Solution:

- ls -ld /usr/local/nagios/var
- sudo chown -R nagios:nagios /usr/local/nagios/var
- sudo chmod -R 755 /usr/local/nagios/var
- sudo systemctl restart nagios

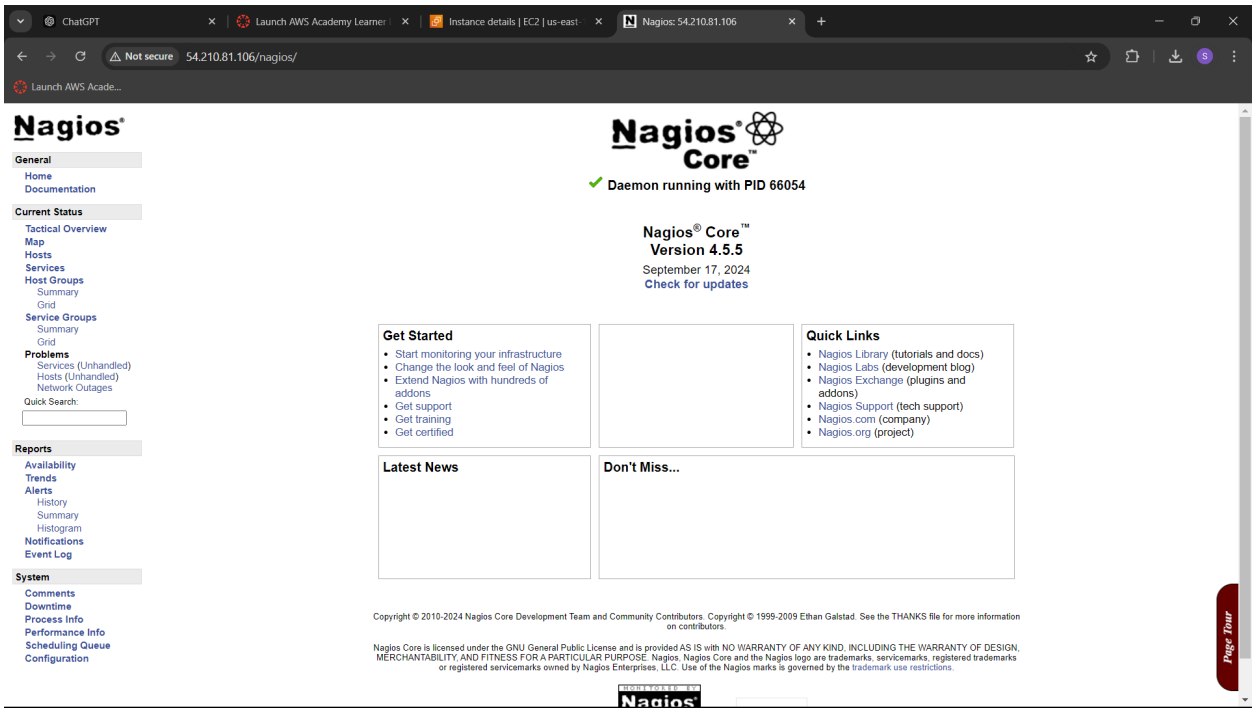
Rerun the command
sudo systemctl status nagios

```
[ec2-user@ip-172-31-83-157 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 Sat 2024-09-28 04:18:59 UTC; 12s ago
       Docs: https://www.nagios.org/documentation
    Process: 66052 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
    Process: 66053 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 66054 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 77ms
     CGroup: /system.slice/nagios.service
             ├─66054 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─66055 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─66056 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─66057 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─66058 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─66059 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: qh: core query handler registered
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: qh: echo service query handler registered
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: qh: help for the query handler registered
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: wproc: Successfully registered manager as @wproc with query handler
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: wproc: Registry request: name=Core Worker 66057;pid=66057
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: wproc: Registry request: name=Core Worker 66058;pid=66058
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: wproc: Registry request: name=Core Worker 66056;pid=66056
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: wproc: Registry request: name=Core Worker 66055;pid=66055
Sep 28 04:18:59 ip-172-31-83-157.ec2.internal nagios[66054]: Successfully launched command file worker with pid 66059
[ec2-user@ip-172-31-83-157 nagios-plugins-2.4.11]$
```

Step 28: Now, go to EC2 instance and click on instance id. Then, click on the copy icon just before the public ip address on public IP.



Step 29: Open a new tab. In the address bar type http://<publicipaddress>/nagios. This would be in the output



Conclusion:

In this experiment, we have learned how to install and configure Nagios Core, Nagios Plugins and NRPE on a Linux machine. We are using an Amazon Linux OS instance configured with the need security rules. We need to make sure that the Nagios-core and Nagios-plugins links that are used are the ones which are up-to dat (wget commands).It is needed to extract and configure these files so that no issues are detected while starting the server. Once all the setup is complete, we can start the nagios server. Using the public IP address of the EC2 instance, we can access the nagios dashboard by navigating that IP to nagios.