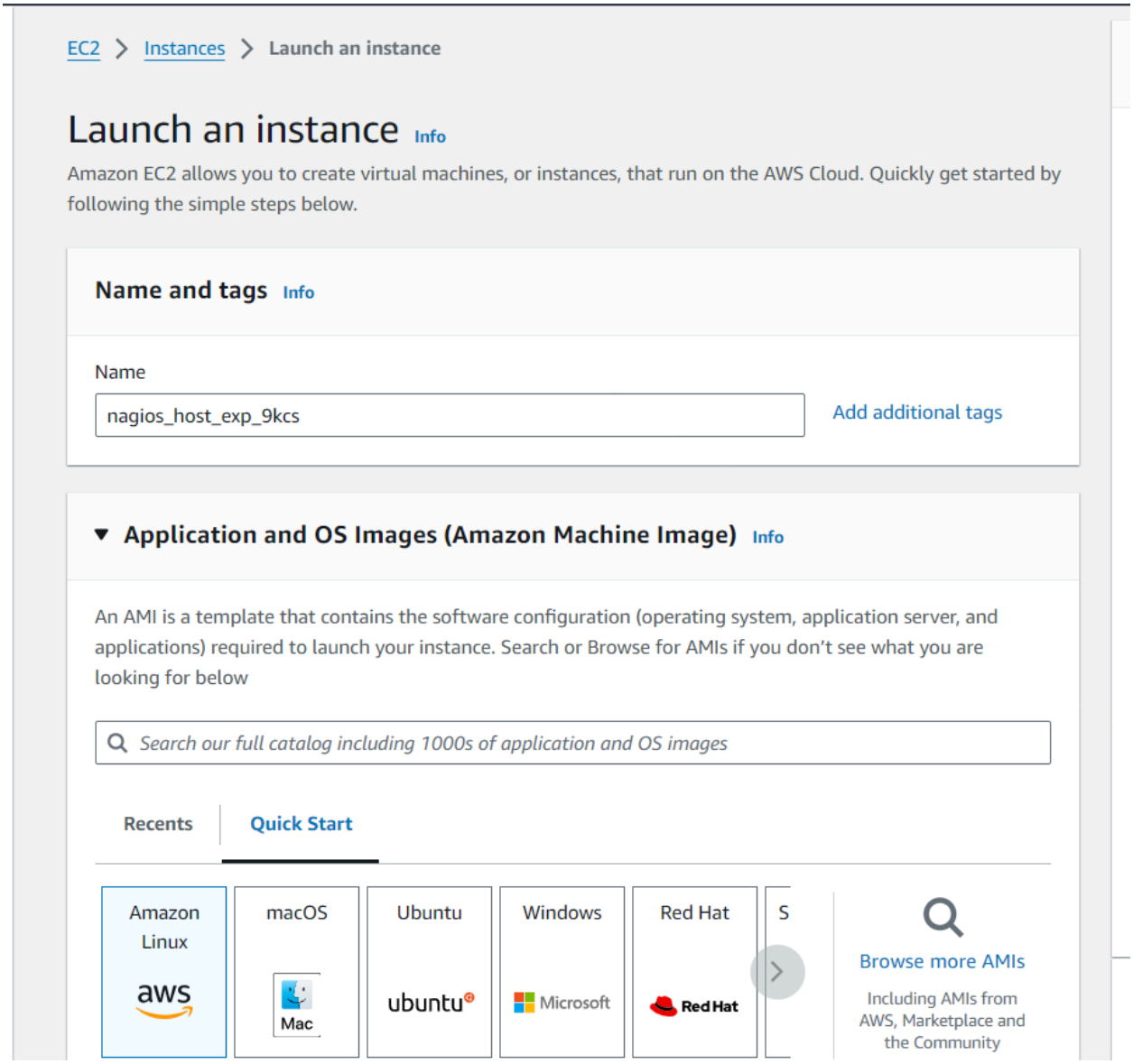
Name : Shubham Jha Div : D15C Roll no:19

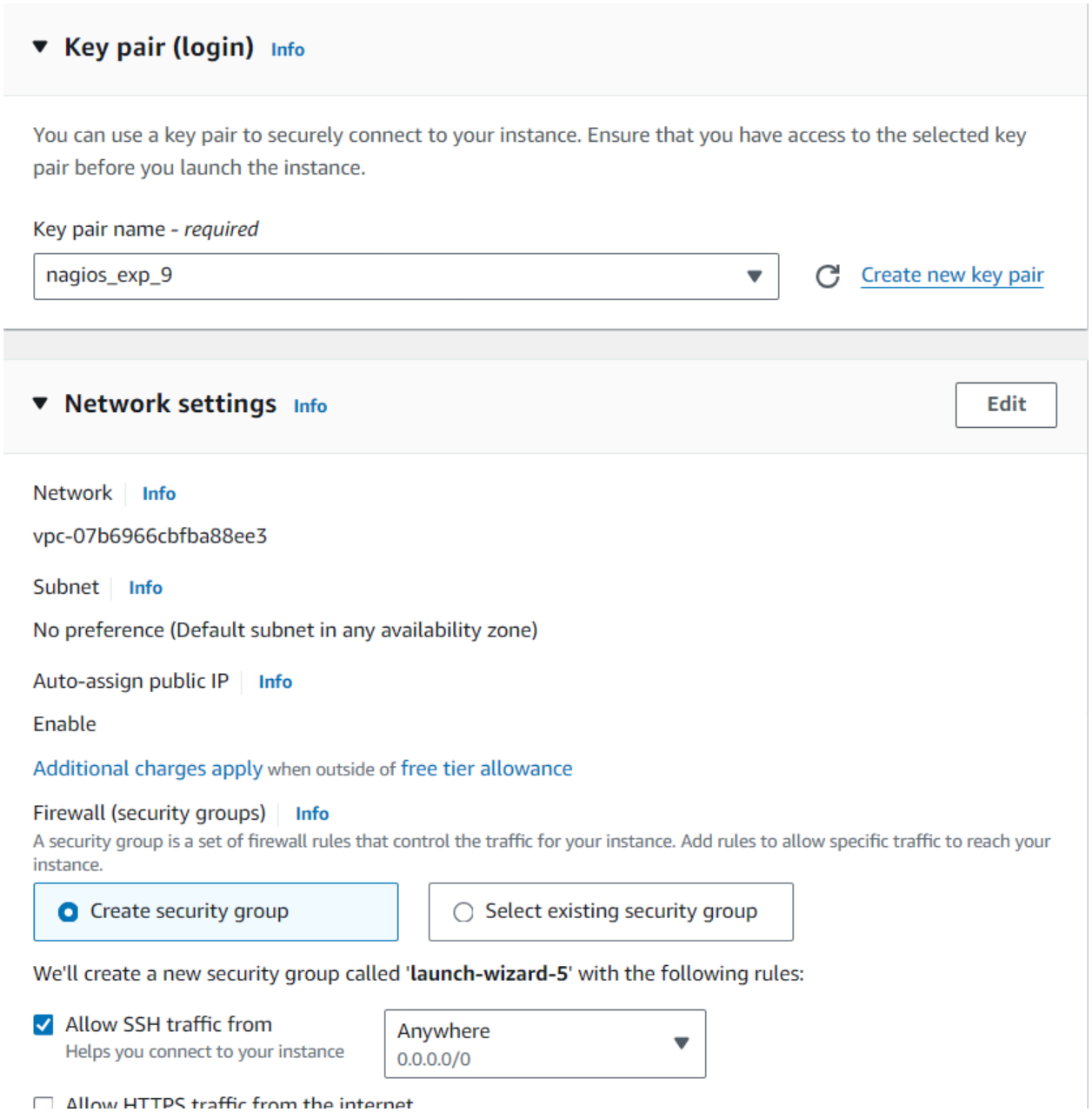
Steps

Launch an ec2 instance
Give name use the default OS



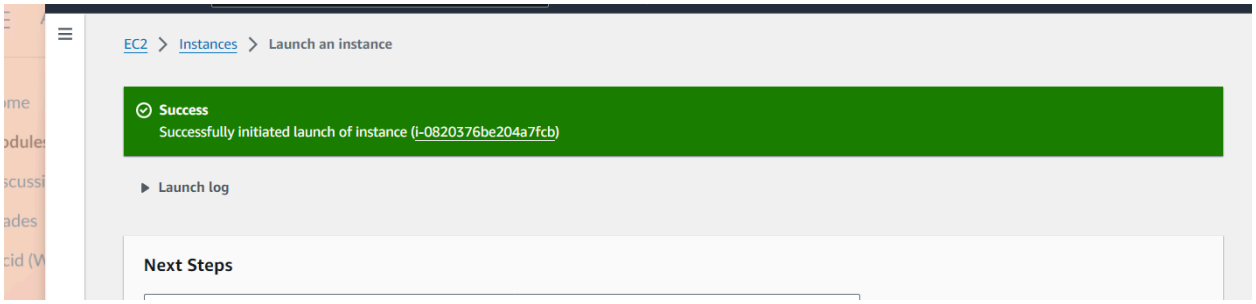
Name : Shubham Jha Div : D15C Roll no:19

Make a key pair and use it.



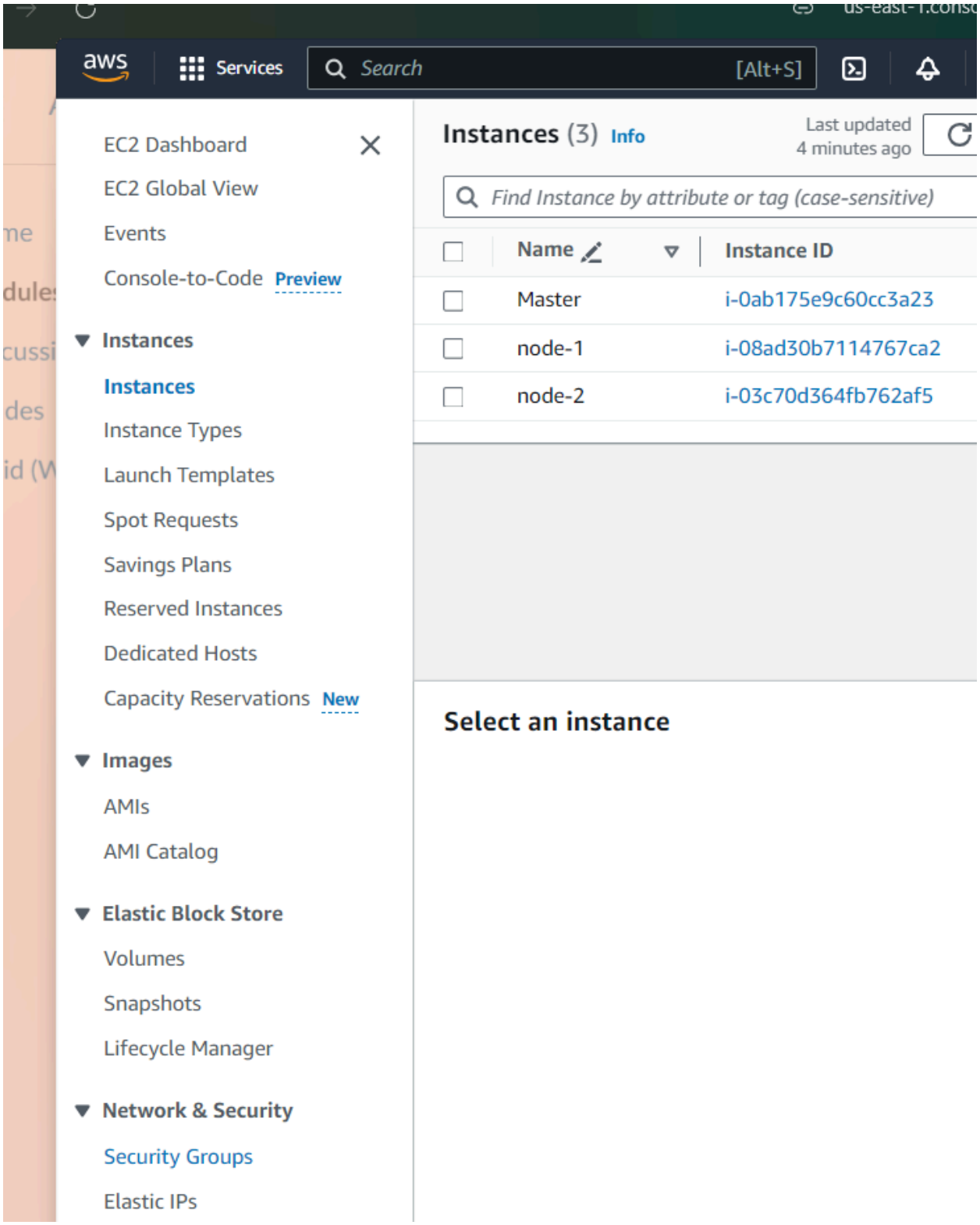
Note the name of the security group that was created for future use:
here it is ‘ launch-wizard-5 ’

Name : Shubham Jha Div : D15C Roll no:19



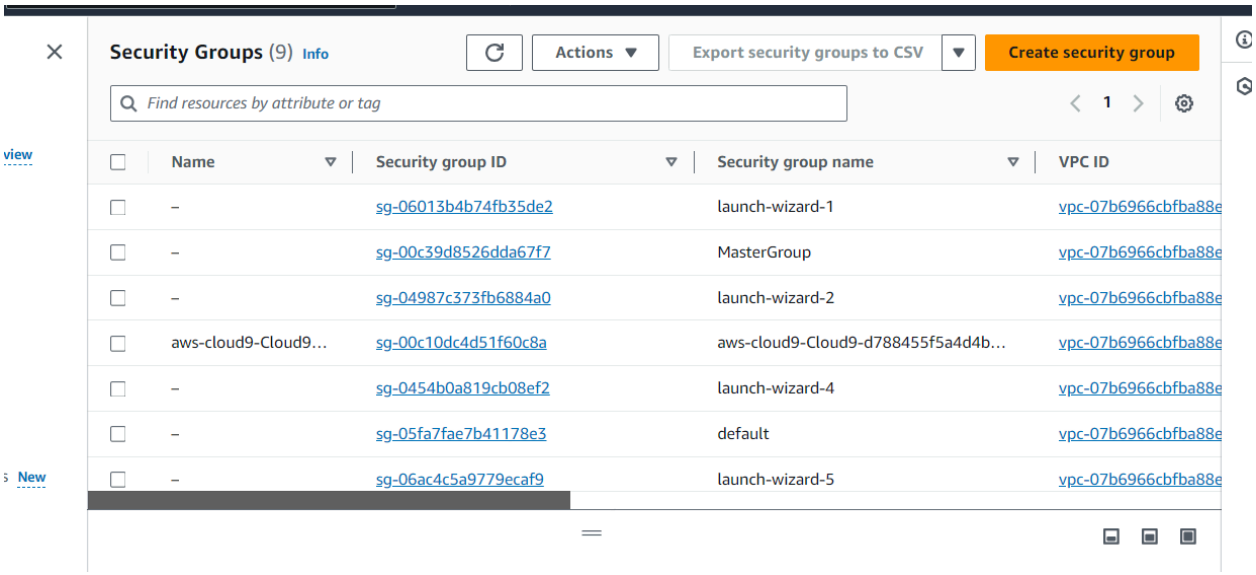
Name : Shubham Jha Div : D15C Roll no:19

go to security groups:

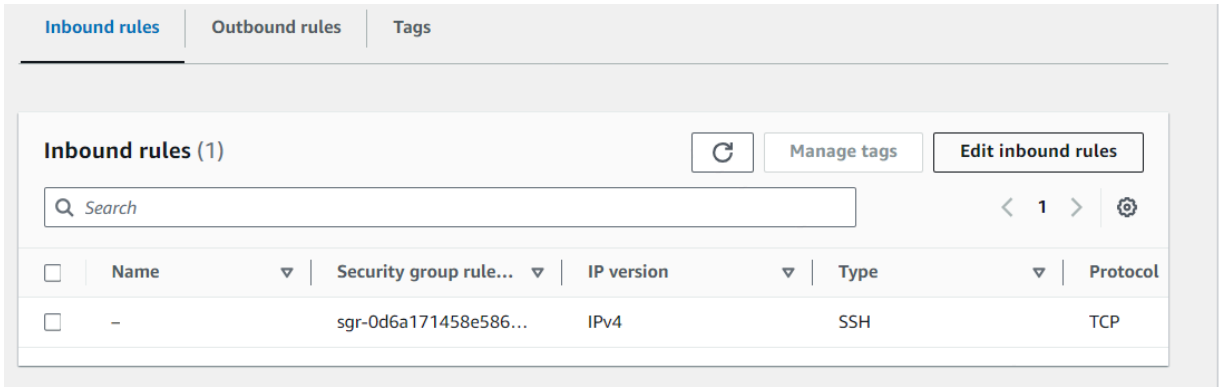


Name : Shubham Jha Div : D15C Roll no:19

click the security group id which was created while you created the ec2 instance of this experiment.

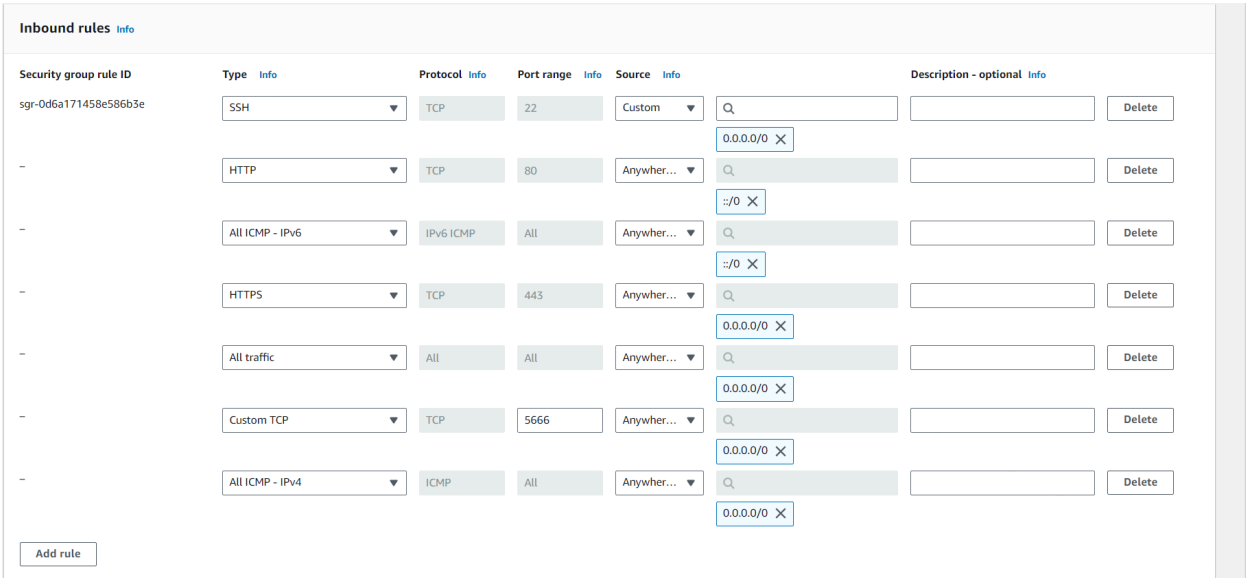


now click on edit inbound rules

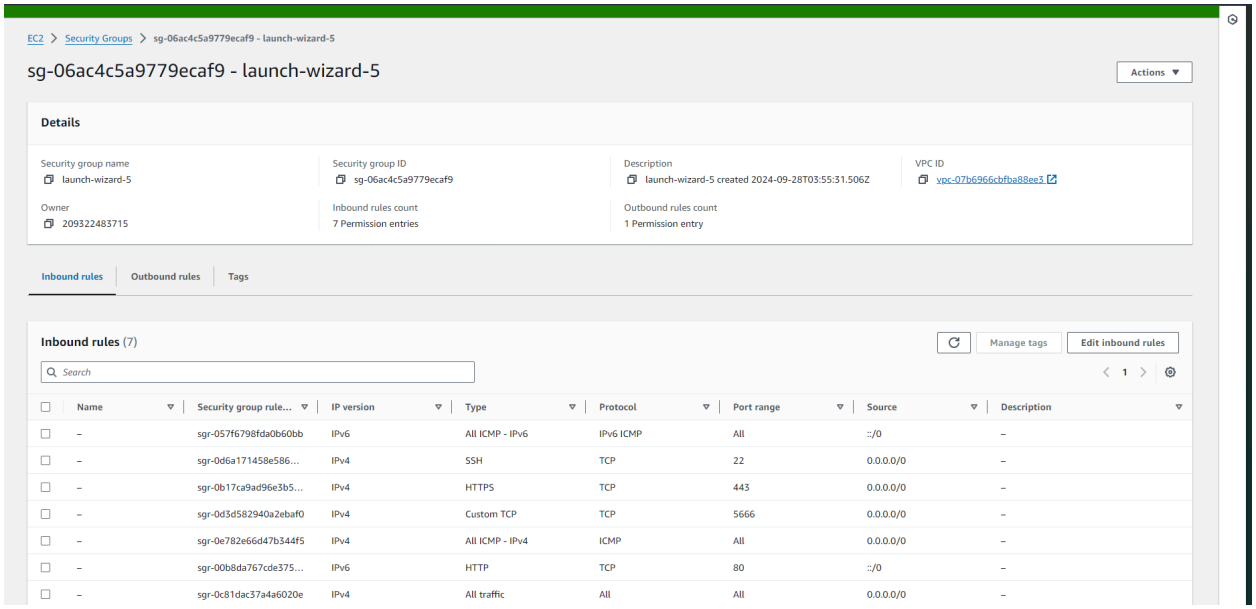


Name : Shubham Jha Div : D15C Roll no:19

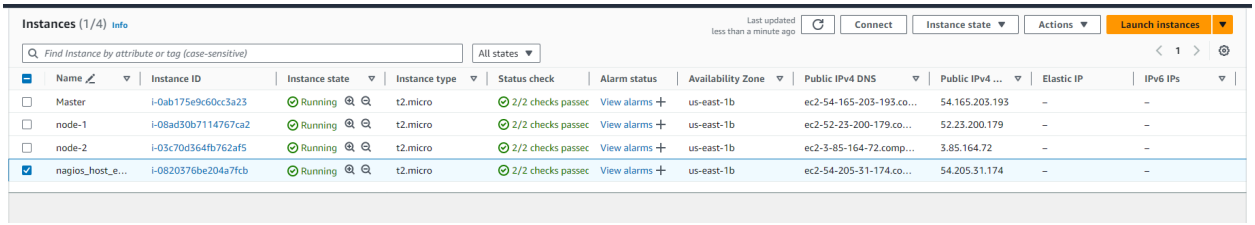
now do the following configurations:
by clicking “add rules”



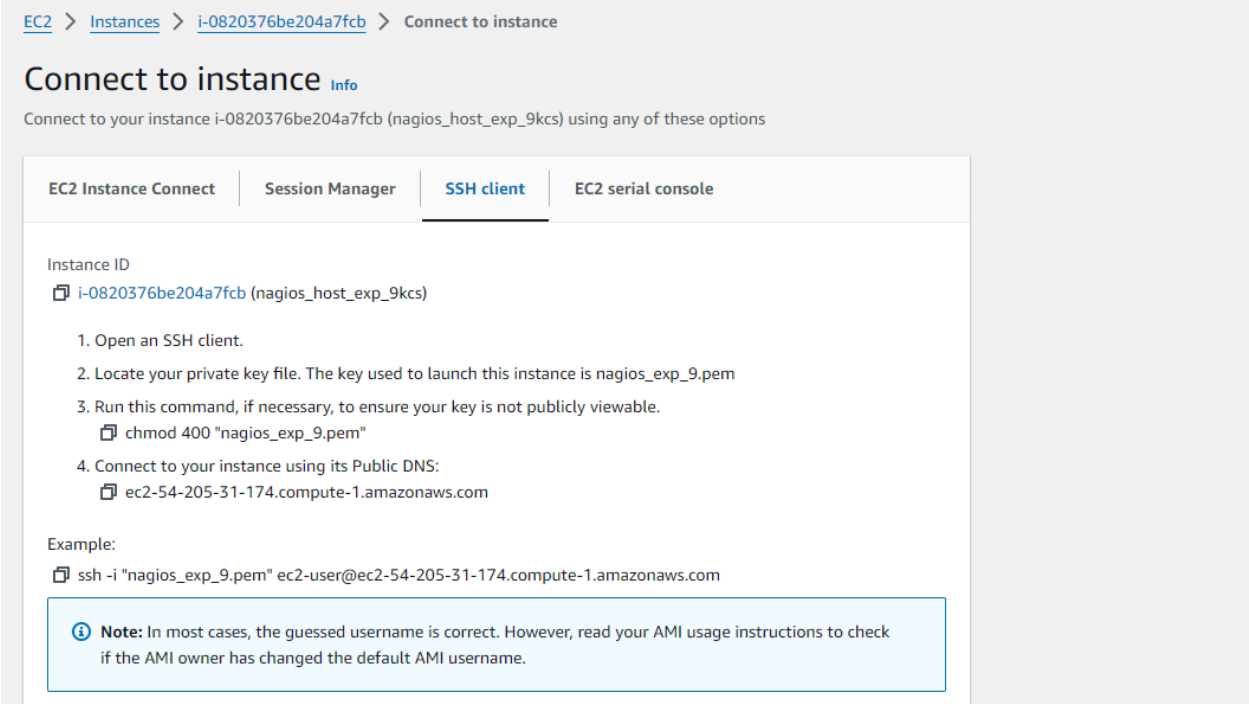
then click on save rules.



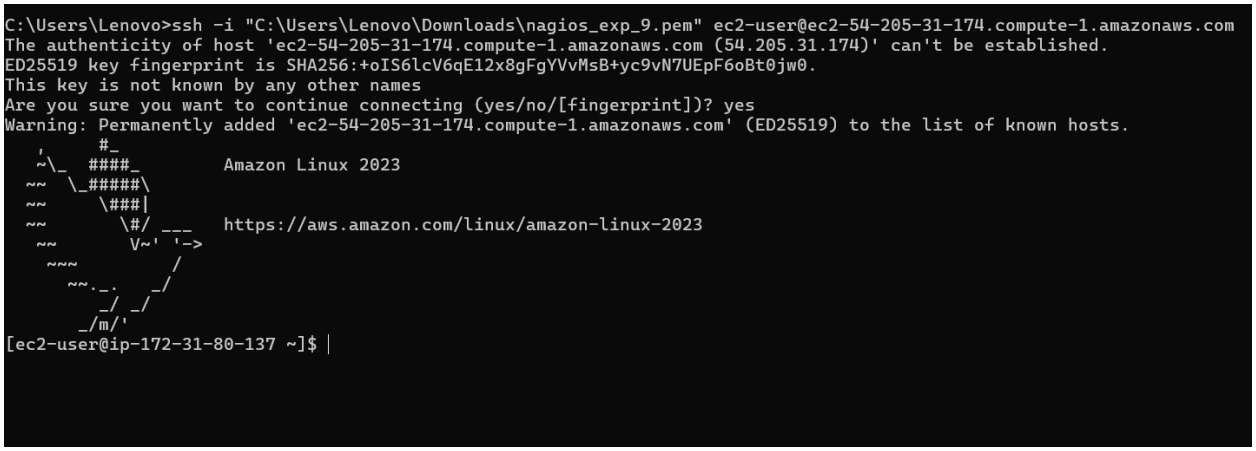
now navigate to instances, click on the instance which was created earlier and click on connect.



now copy the ssh command and just replace the .pem file with its actual location in your computer.



paste the command in your terminal and enter after replacing the .pem file with its actual location in your system.



Name : Shubham Jha Div : D15C Roll no:19

now paste the following commands in your connected terminal:

sudo yum update

```
       _/m/'
[ec2-user@ip-172-31-80-137 ~]$ sudo yum update
Last metadata expiration check: 2:21:45 ago on Sat Sep 28 03:59:04 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-80-137 ~]$
```

sudo yum install httpd php

```
[ec2-user@ip-172-31-80-137 ~]$ sudo yum install httpd php
Last metadata expiration check: 2:22:53 ago on Sat Sep 28 03:59:04 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture       Version                 Repository
================================================================================
Installing:
 httpd                   x86_64             2.4.62-1.amzn2023       amazonlinux
 php8.3                  x86_64             8.3.10-1.amzn2023.0.1   amazonlinux
Installing dependencies:
 apr                     x86_64             1.7.2-2.amzn2023.0.2    amazonlinux
 apr-util                x86_64             1.6.3-1.amzn2023.0.1    amazonlinux
 generic-logos-httpd     noarch             18.0.0-12.amzn2023.0.3  amazonlinux
 httpd-core              x86_64             2.4.62-1.amzn2023       amazonlinux
 httpd-filesystem        noarch             2.4.62-1.amzn2023       amazonlinux
```

(type y when prompted)

sudo yum install gcc glibc glibc-common

```
Dependencies resolved.
================================================================================
 Package                 Architecture       Version                 Repository
================================================================================
Installing:
 gcc                     x86_64             11.4.1-2.amzn2023.0.2   amazonlinux
Installing dependencies:
 annobin-docs            noarch             10.93-1.amzn2023.0.1    amazonlinux
 annobin-plugin-gcc      x86_64             10.93-1.amzn2023.0.1    amazonlinux
 cpp                     x86_64             11.4.1-2.amzn2023.0.2   amazonlinux
 gc                      x86_64             8.0.4-5.amzn2023.0.2    amazonlinux
 glibc-devel             x86_64             2.34-52.amzn2023.0.11   amazonlinux
 glibc-headers-x86       noarch             2.34-52.amzn2023.0.11   amazonlinux
 guile22                 x86_64             2.2.7-2.amzn2023.0.3    amazonlinux
 kernel-headers          x86_64             6.1.109-118.189.amzn2023 amazonlinux
 libmpc                  x86_64             1.2.1-2.amzn2023.0.2    amazonlinux
 libtool-ltdl            x86_64             2.4.7-1.amzn2023.0.3    amazonlinux
 libxcrypt-devel         x86_64             4.4.33-7.amzn2023       amazonlinux
 make                    x86_64             1:4.3-5.amzn2023.0.2    amazonlinux

Transaction Summary
================================================================================
Install  13 Packages

Total download size: 52 M
Installed size: 168 M
Is this ok [y/N]: y
```

sudo yum install gd gd-devel

Name : Shubham Jha Div : D15C Roll no:19

```
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

omplete!
ec2-user@ip-172-31-80-137 ~]$
```

sudo adduser -m nagios
sudo passwd nagios

```
Complete!
[ec2-user@ip-172-31-80-137 ~]$ 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.
[ec2-user@ip-172-31-80-137 ~]$
```

(add a password here)

sudo groupadd nagcmd

```
[ec2-user@ip-172-31-80-137 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-80-137 ~]$
```

sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache

```
[ec2-user@ip-172-31-80-137 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-80-137 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-80-137 ~]$
```

mkdir ~/downloads
cd ~/downloads

Name : Shubham Jha Div : D15C Roll no:19

```
[ec2-user@ip-172-31-80-137 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-80-137 downloads]$
```

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

```
cd ~/downloads
[ec2-user@ip-172-31-80-137 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-09-28 06:27:51--  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  5.30MB/s    in 0.4s

2024-09-28 06:27:52 (5.30 MB/s) - ‘nagios-4.5.5.tar.gz’ saved [2065473/2065473]

[ec2-user@ip-172-31-80-137 downloads]$
```

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

```
[ec2-user@ip-172-31-80-137 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-09-28 06:28:14--  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  5.90MB/s    in 0.4s

2024-09-28 06:28:15 (5.90 MB/s) - ‘nagios-plugins-2.4.11.tar.gz’ saved [2753049/2753049]

[ec2-user@ip-172-31-80-137 downloads]$
```

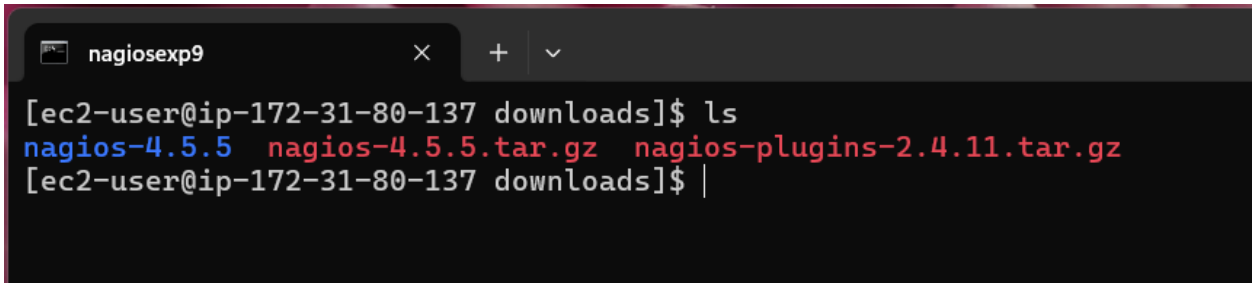
tar zxvf nagios-4.5.5.tar.gz

```
[ec2-user@ip-172-31-80-137 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
```

Now we have to first navigate to the nagios-4.5.5 folder in downloads.

- **commands to enter:**
  ls (verify whether nagios-4.5.5 exists)

Name : Shubham Jha Div : D15C Roll no:19

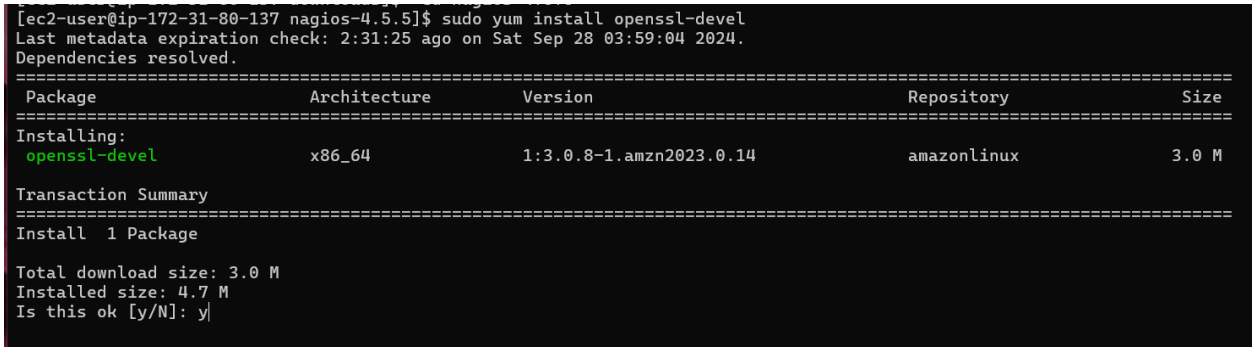


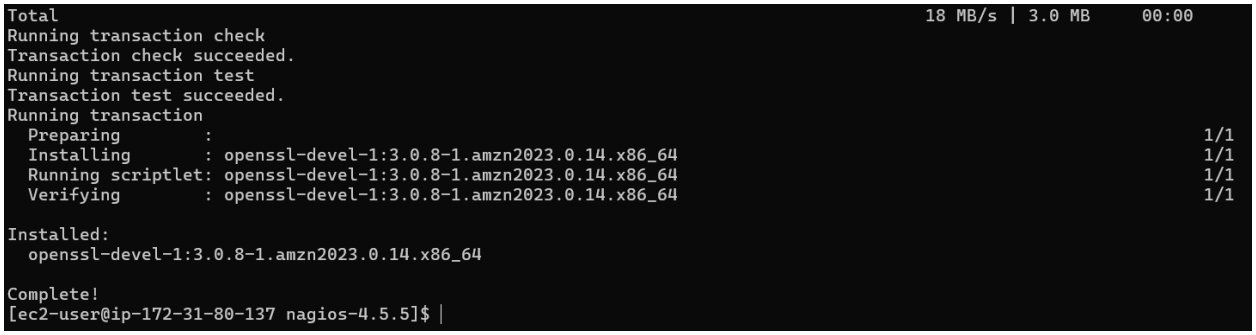
cd nagios-4.5.5

we now have to install openssl dev library

commands to enter:

sudo yum install openssl-devel

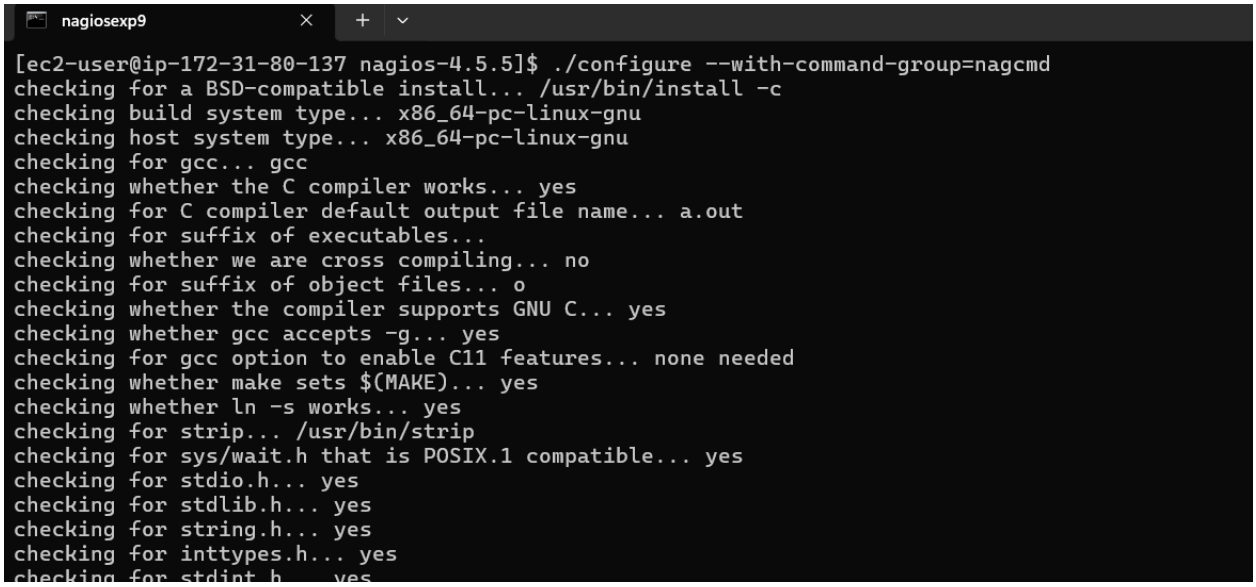




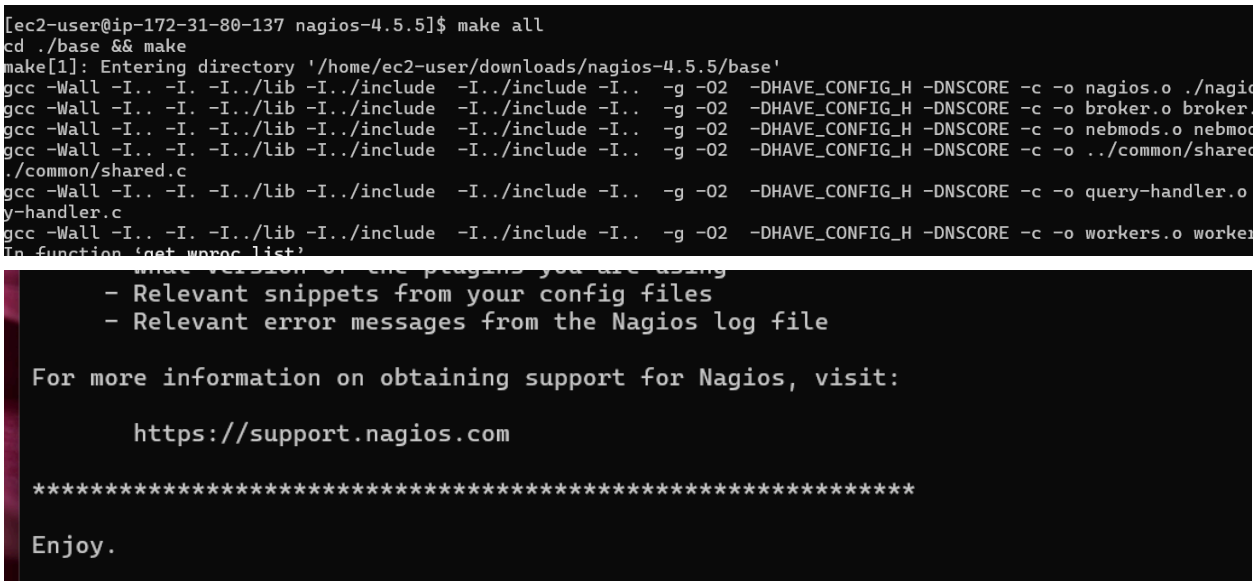
Then finally we can run the commands like usual.

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

Name : Shubham Jha Div : D15C Roll no:19



make all



sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode

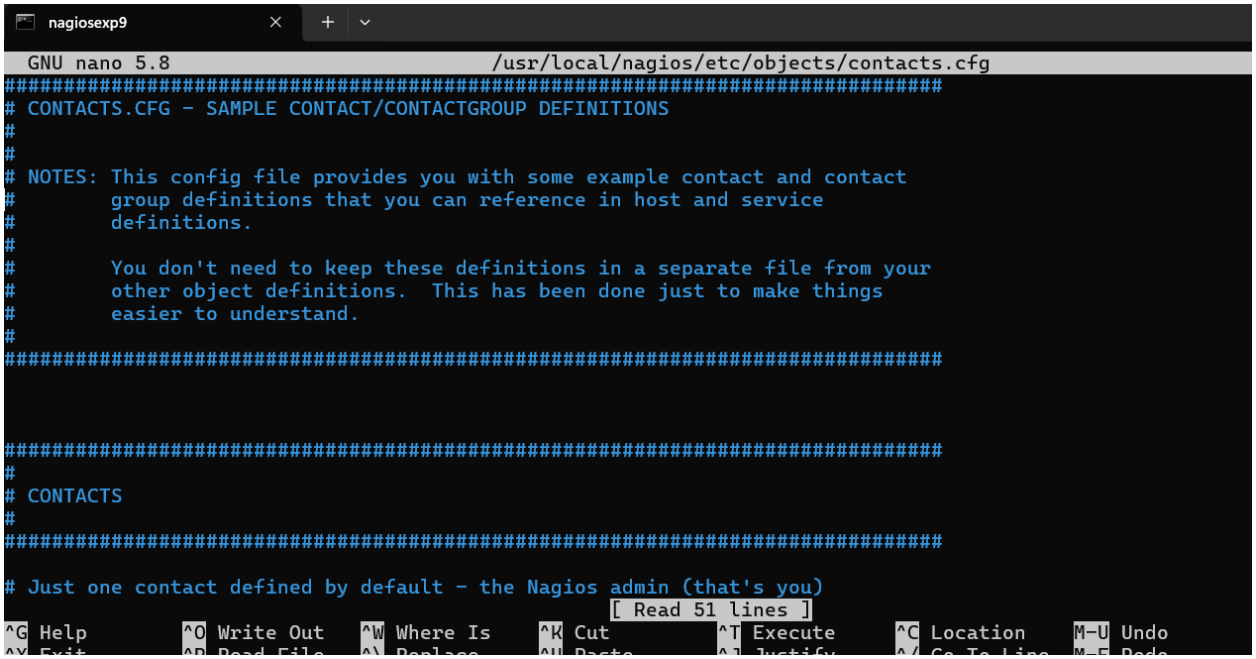
Name : Shubham Jha Div : D15C Roll no:19

```
[ec2-user@ip-172-31-80-137 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
for file in *.cgi; do \
```

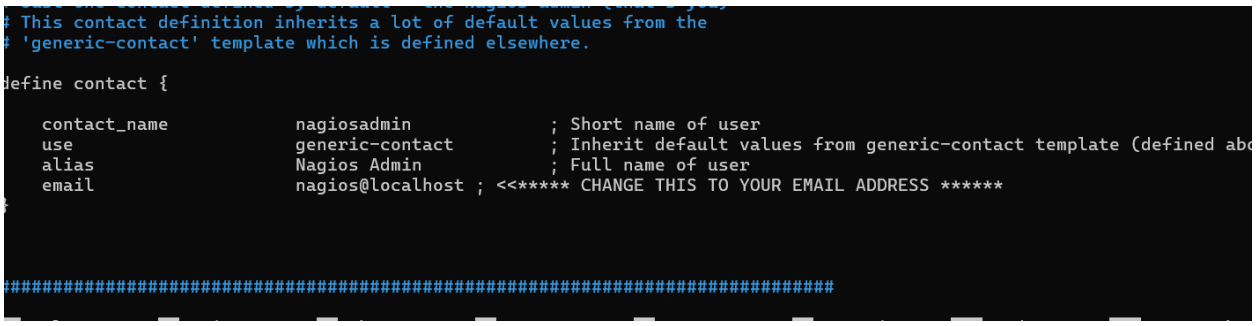
Now the next command will take us to nano editor:

---

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



navigate down to email: and change it to your email address.



```
# 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 (define
    alias                   Nagios Admin            ; Full name of user
    email                   2022.shubham.jha@ves.ac.in ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS ******
}



###############################################################################
#
# CONTACT GROUPS

^G Help       ^O Write Out  ^W Where Is   ^K Cut         ^T Execute    ^C Location   M-U Undo      M-A Set M
```

press Ctrl+O and then enter.
then press Ctrl +X

```
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$
```

---

sudo make install-webconf

```
[ec2-user@ip-172-31-80-137 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-172-31-80-137 nagios-4.5.5]$
```

Adding password for nagios admin

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

```
[ec2-user@ip-172-31-80-137 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
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$
```

sudo service httpd restart

Name : Shubham Jha Div : D15C Roll no:19

```
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$
```

cd ~/downloads
tar zxvf nagios-plugins-2.4.11.tar.gz

```
[ec2-user@ip-172-31-80-137 downloads]$ 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/
```

cd nagios-plugins-2.4.11
./configure --with-nagios-user=nagios --with-nagios-group=nagios

```
nagiosexp9
[ec2-user@ip-172-31-80-137 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)
```

Name : Shubham Jha Div : D15C Roll no:19

make
sudo make install

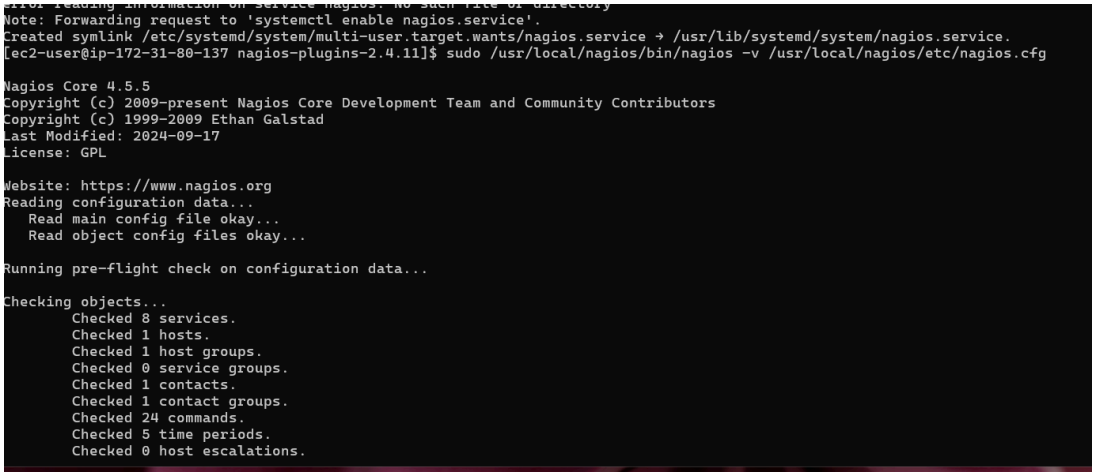
```
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/plugins-root'
aking install in po
ake[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/po'
usr/bin/mkdir -p /usr/local/nagios/share
nstalling fr.gmo as /usr/local/nagios/share/locale/fr/LC_MESSAGES/nagios-plugins.mo
nstalling de.gmo as /usr/local/nagios/share/locale/de/LC_MESSAGES/nagios-plugins.mo
f test "nagios-plugins" = "gettext-tools"; then \
 /usr/bin/mkdir -p /usr/local/nagios/share/gettext/po; \
 for file in Makefile.in.in remove-potcdate.sin    Makevars.template; do \
   /usr/bin/install -c -o nagios -g nagios -m 644 ./$file \
                    /usr/local/nagios/share/gettext/po/$file; \
 done; \
 for file in Makevars; do \
   rm -f /usr/local/nagios/share/gettext/po/$file; \
 done; \
lse \
 : ; \
i
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/po'
ake[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[2]: Nothing to be done for 'install-exec-am'.
ake[2]: Nothing to be done for 'install-data-am'.
ake[2]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
```

sudo chkconfig --add nagios
sudo chkconfig nagios on

```
[ec2-user@ip-172-31-80-137 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.
[ec2-user@ip-172-31-80-137 nagios-plugins-2.4.11]$
```

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

Name : Shubham Jha Div : D15C Roll no:19



If this command is giving 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) Error processing main config file!**)**

The solution:

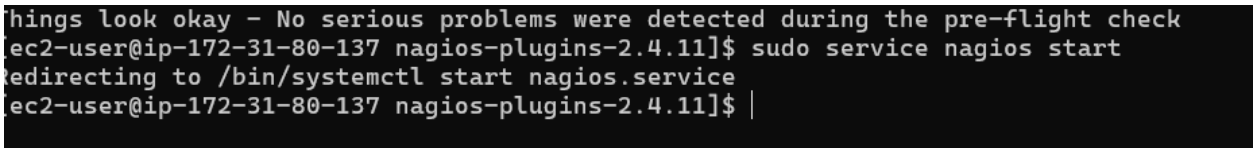
Create the missing directory, set the permissions, verify it.

sudo mkdir -p /usr/local/nagios/var/spool/checkresults (this is for creation)
sudo chown nagios:nagios /usr/local/nagios/var/spool/checkresults
sudo chmod 775 /usr/local/nagios/var/spool/checkresults (this is for permissions)

Now rerun the commmad (also given below) and continue:

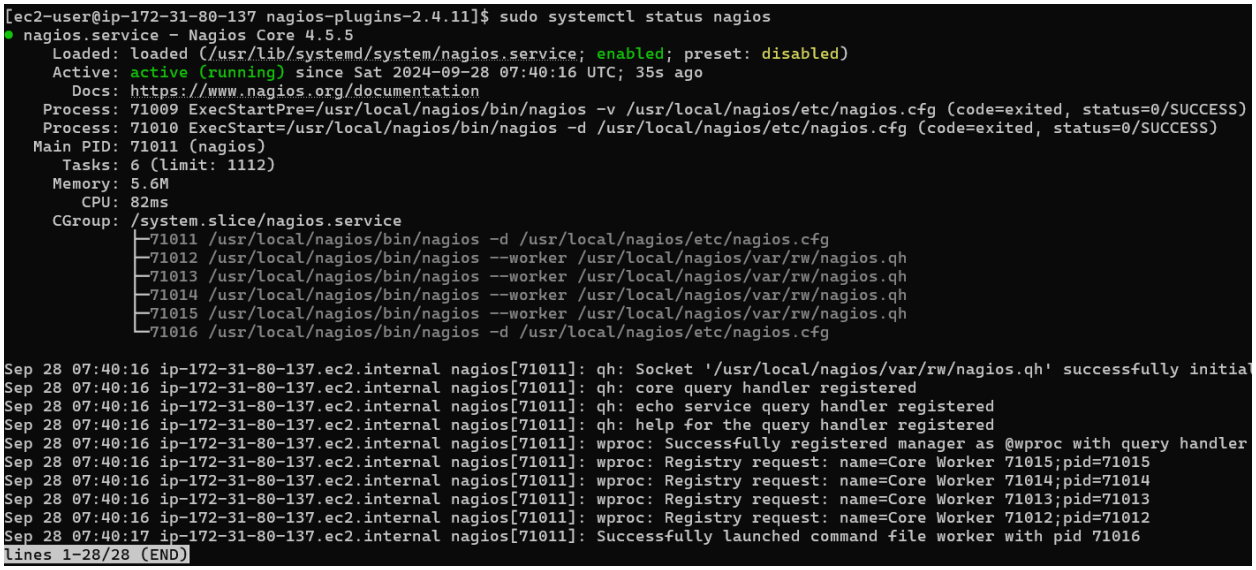
sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

sudo service nagios start



---

sudo systemctl status nagios



(ignore if no error was found)
Again if this is giving an error then it is primarily because Nagios monitoring tool is unable to create or write to a temporary file in the "`/usr/local/nagios/var/`"

To debug it lets start by checking the permissions:
ls -ld /usr/local/nagios/var

Changing the ownership
sudo chown -R nagios:nagios /usr/local/nagios/var

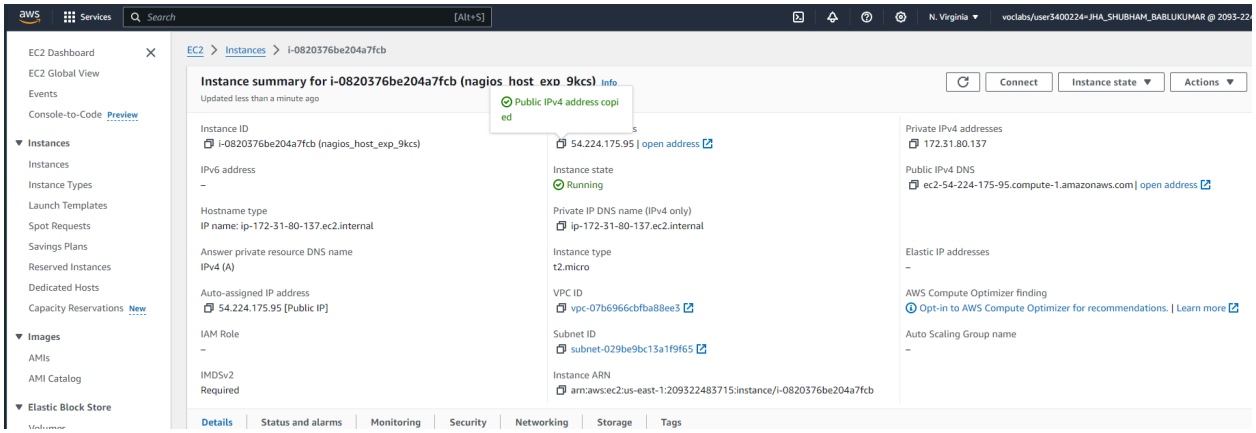
Modify permissions
sudo chmod -R 755 /usr/local/nagios/var

Restart Nagios service
sudo systemctl restart nagios

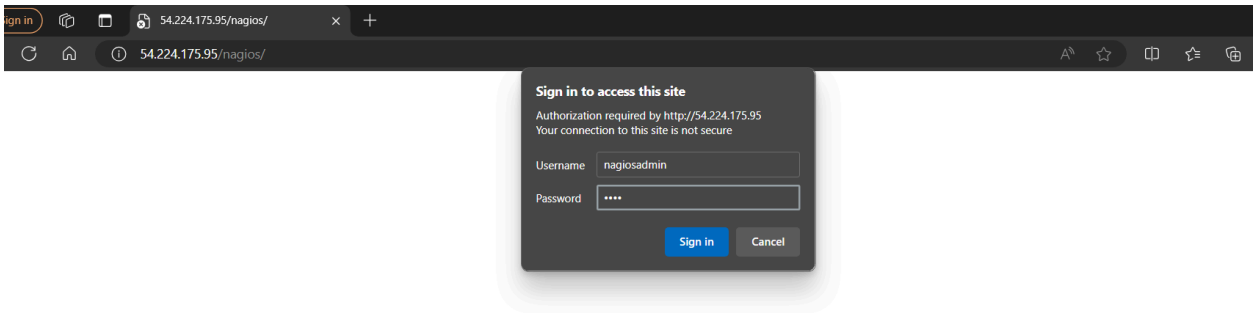
check status of nagios, Rerun the command **(the command which gave the recent error)**
sudo systemctl status nagios

Name : Shubham Jha Div : D15C Roll no:19

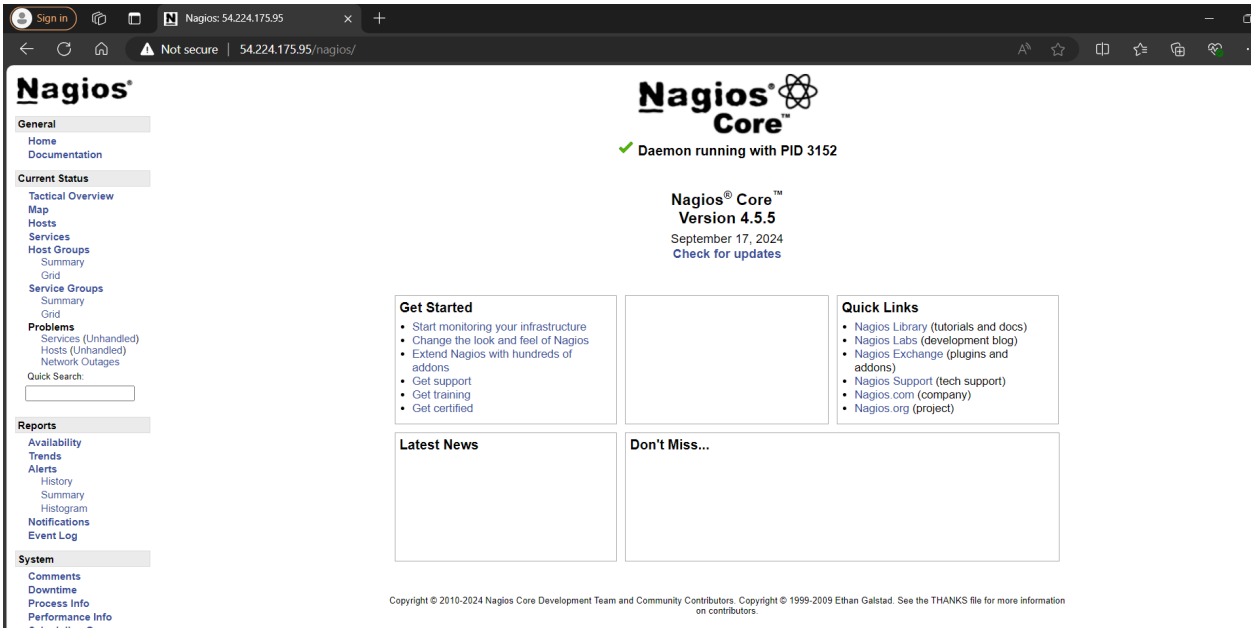
Now, go to EC2 instance and click on instance id. Then, click on the copy icon just before the public ip address on public IP.



Enter the username password set above. (in the section of adding password for nagios admin)



Name : Shubham Jha Div : D15C Roll no:19



Conclusion:

Setting up Nagios on an EC2 instance was a rewarding yet challenging experience for me. I began by launching an instance using the default operating system and configuring it to monitor my network. The installation process went smoothly at first; I installed essential packages, created users, and configured Nagios as planned.

However, I encountered a few hurdles along the way. One significant issue arose when the Apache server was not running, which prevented me from accessing the Nagios web interface. After some troubleshooting, I realized that restarting the Apache service was necessary to resolve this.

Additionally, I faced permission issues that initially hindered Nagios from creating or writing to temporary files. By checking the ownership and permissions of the necessary directories, I managed to address this issue effectively.