Experiment No:10

Aim: To perform Port, Service monitoring, and Windows/Linux server monitoring using Nagios.

Prerequisites:
AWS Academy or Personal account.
Nagios Server running on Amazon Linux Machine. (Refer Experiment No 9)

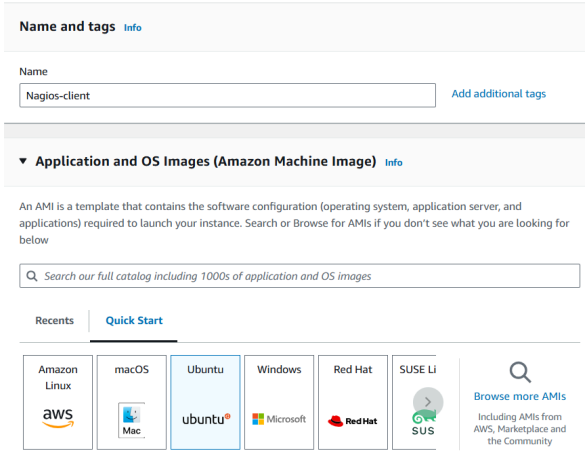
Monitoring Using Nagios:

1. Firstly,Confirm nagios Host is running or not by checking its status by following command.
   sudo systemctl status nagios

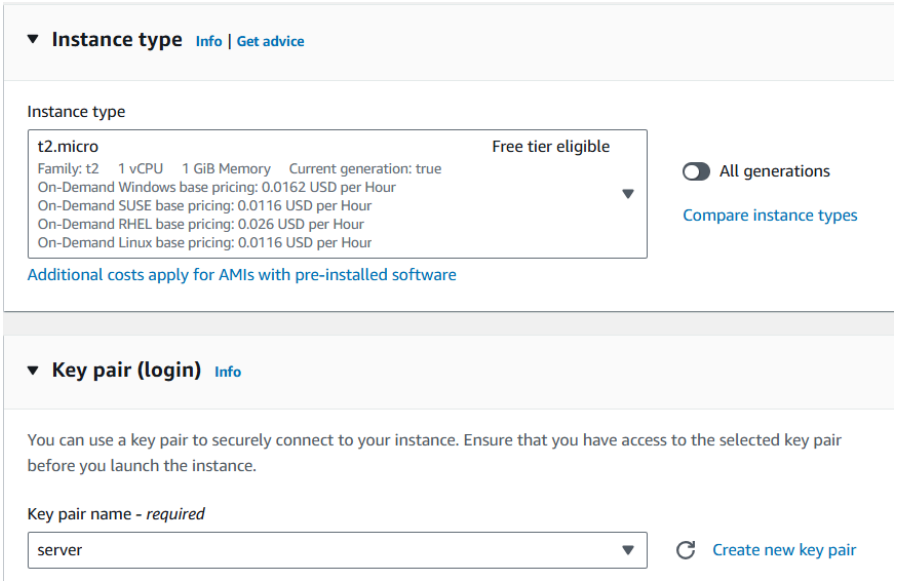
```
[ec2-user@ip-172-31-39-90 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; disabled; preset: disabled)
     Active: active (running) since Fri 2024-10-04 04:14:29 UTC; 9min ago
       Docs: https://www.nagios.org/documentation
    Process: 75298 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
    Process: 75299 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 75300 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 164ms
     CGroup: /system.slice/nagios.service
             ├─75300 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─75301 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─75302 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─75303 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─75304 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─75305 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 04:17:36 ip-172-31-39-90.ec2.internal nagios[75300]: SERVICE ALERT: localhost;Root Partition;CRITICAL;HARD;1;(No output on st>
Oct 04 04:18:14 ip-172-31-39-90.ec2.internal nagios[75300]: SERVICE ALERT: localhost;SSH;CRITICAL;HARD;1;(No output on stdout) stder>
Oct 04 04:18:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;SOFT;5;(No output on stdout) stderr: execvp(/>
Oct 04 04:18:51 ip-172-31-39-90.ec2.internal nagios[75300]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;HARD;1;(No output on stdout>
Oct 04 04:19:29 ip-172-31-39-90.ec2.internal nagios[75300]: SERVICE ALERT: localhost;Total Processes;CRITICAL;HARD;1;(No output on s>
Oct 04 04:19:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;SOFT;6;(No output on stdout) stderr: execvp(/>
Oct 04 04:20:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;SOFT;7;(No output on stdout) stderr: execvp(/>
Oct 04 04:21:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;SOFT;8;(No output on stdout) stderr: execvp(/>
Oct 04 04:22:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;SOFT;9;(No output on stdout) stderr: execvp(/>
Oct 04 04:23:29 ip-172-31-39-90.ec2.internal nagios[75300]: HOST ALERT: localhost;DOWN;HARD;10;(No output on stdout) stderr: execvp(>
lines 1-28/28 (END)
```

You can now proceed ahead if you get the above message/output.If not then again create a instance (Refer Experiment No 9)

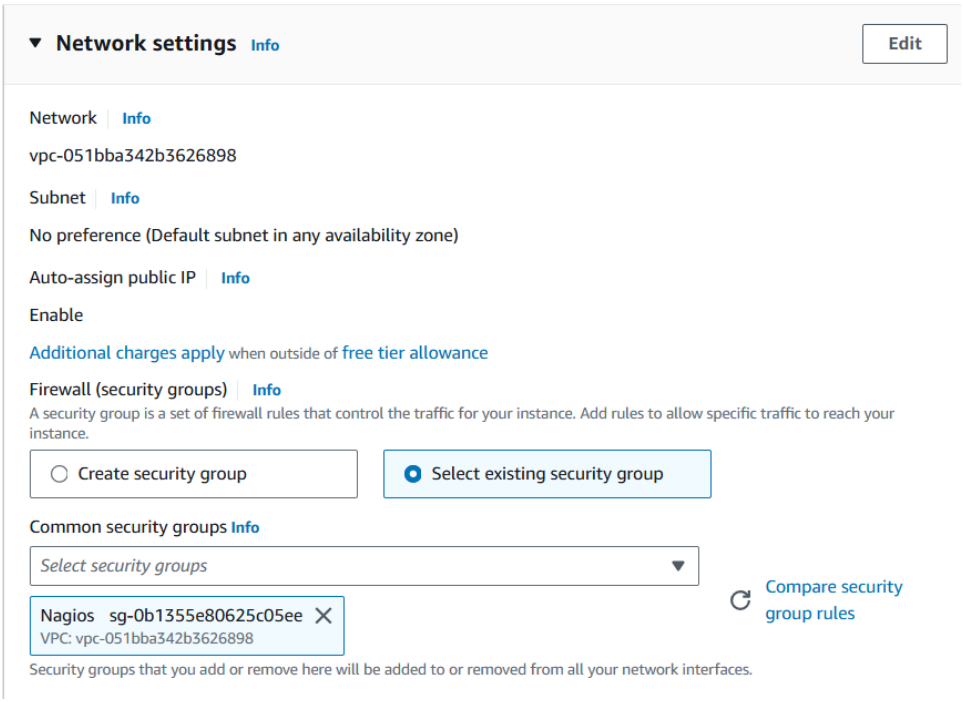
2. **Creation an EC-2 Instance:**Create an EC-2 Instance of t2.micro type on ubuntu in AWS.



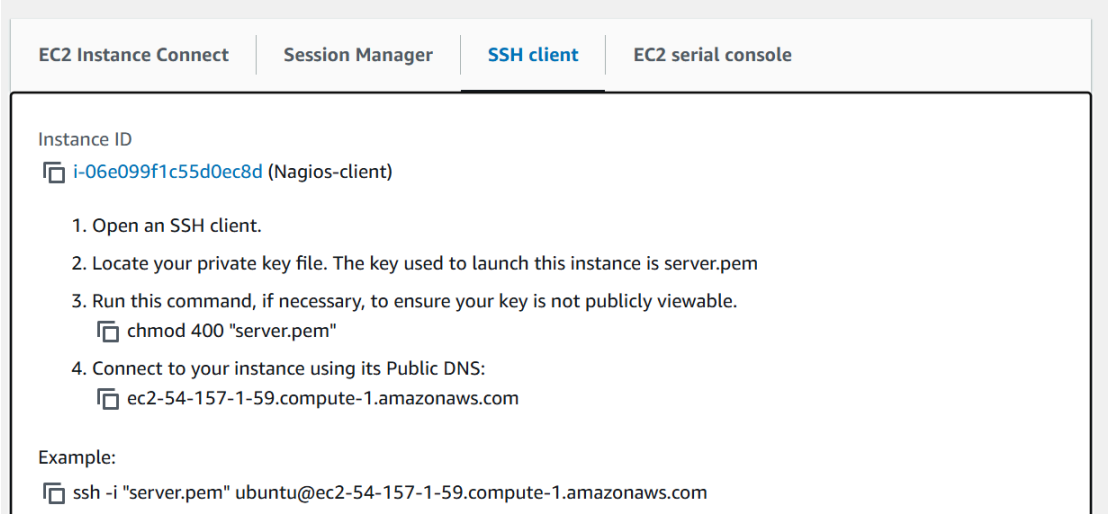
3. Then select the key pair that you have created and used in “nagios-host” EC-2 instance.



4. Then select the security group that we have created in nagios host.here I have given name to security groups as Nagios.



5. **Making the connection:**Then after instance get created successfully then click on connect then go to SSH client section there you will see connection command **ssh -i “key_pair.pem” ubuntu@ec2-Public_ip.compute-1.amazonaws.com**



Copy paste this command on command prompt.

```
PS C:\Users\Sadneya\downloads> ssh -i "server.pem" ec2-user@ec2-3-81-91-101.compute-1.amazonaws.com
The authenticity of host 'ec2-3-81-91-101.compute-1.amazonaws.com (3.81.91.101)' can't be established.
ED25519 key fingerprint is SHA256:6UVLjB6FbGB7A92sIEobs4886tozb5yML0ekn5Xzfrw.
This host key is known by the following other names/addresses:
    C:\Users\Sadneya/.ssh/known_hosts:68: ec2-18-232-155-134.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-81-91-101.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Fri Oct  4 03:36:10 2024 from 125.99.93.18
```

****Now perform Following commands on Nagios-host****

6. Now on the server Nagios-host run the following command.

ps -ef | grep nagios

```
[ec2-user@ip-172-31-39-90 ~]$ ps -ef | grep nagios
ec2-user    2377    2350  0 09:15 pts/0    00:00:00 grep --color=auto nagios
```

7. Now become the root user by sudo su and create the directories.

sudo su
mkdir /usr/local/nagios/etc/objects/monitorhosts
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts

```
[ec2-user@ip-172-31-39-90 ~]$ sudo su
[root@ip-172-31-39-90 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-39-90 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

8. Copy the localhost.cfg file to linuxserver.cfg by following command.

cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

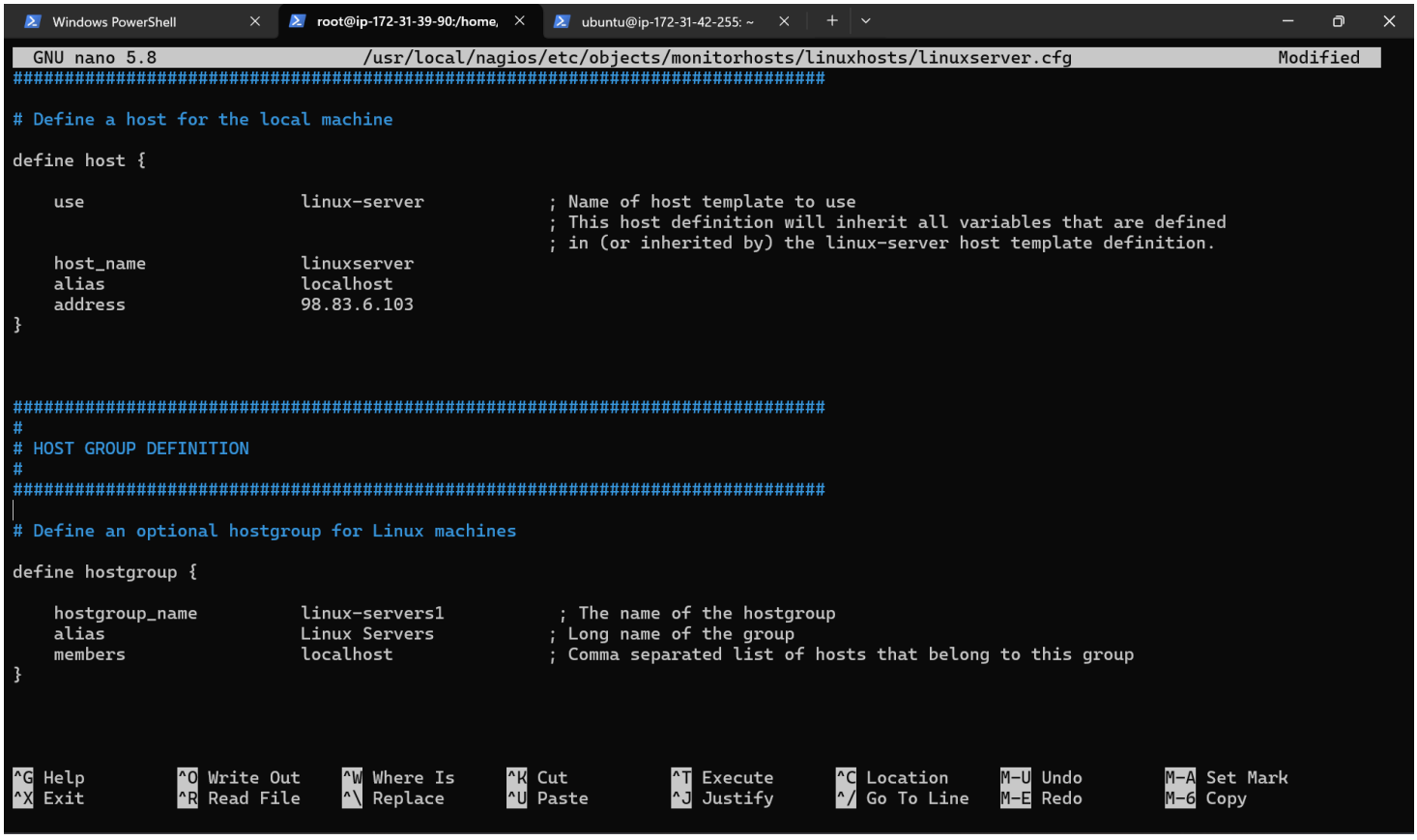
```
[root@ip-172-31-39-90 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linux
hosts/linuxserver.cfg
```

9. Open the linuxserver.cfg by nano command and then make following changes.

nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

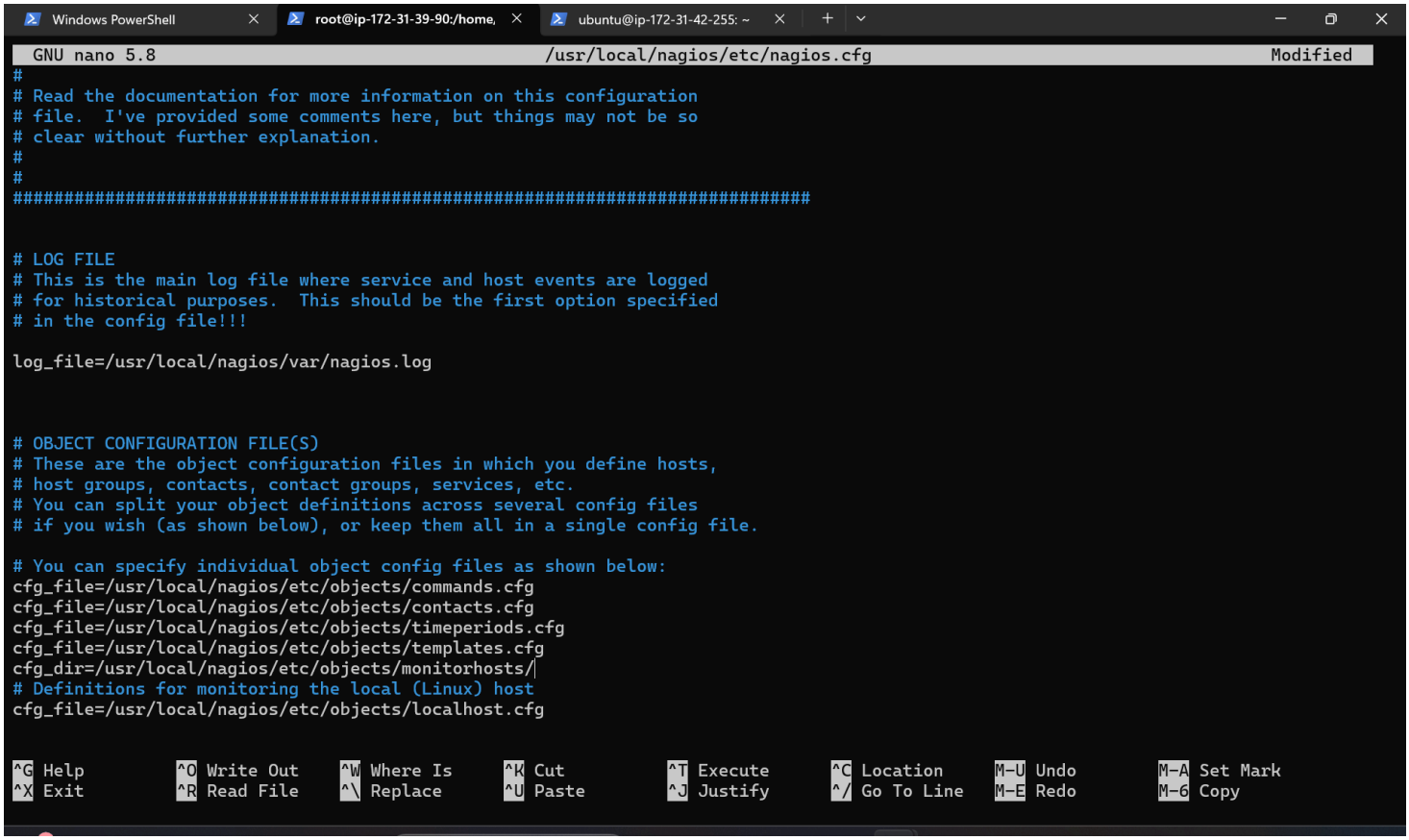
```
[root@ip-172-31-39-90 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-39-90 ec2-user]#
```

Now change the hostname to **linuxserver**. Then Change the address to the public IP of your Linux client. Now Set hostgroup_name to **linux-servers1**.

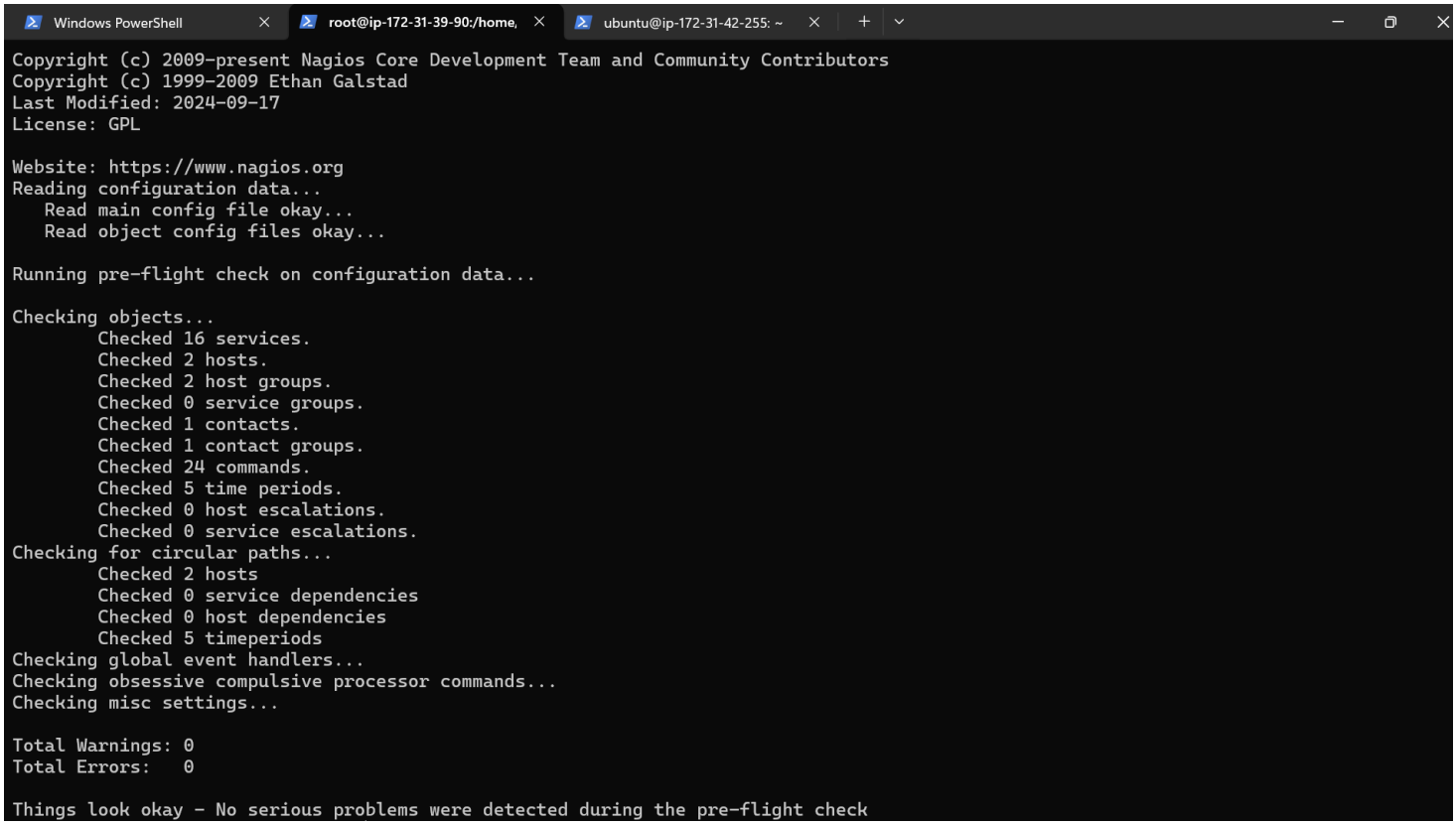


Make the hostname changes throughout in Host,Host Group,HTTP,SSH.

10. Now update the Nagios.config file add the following line in it.
    cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
    Then execute the following command : **nano /usr/local/nagios/etc/nagios.cfg**



11. Now verify the configuration by the following command.
/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg



12. Now restart the services of nagios by running the following command.

service nagios restart

```
[root@ip-172-31-39-90 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
```

****Now perform Following commands on Nagios-host****

13. Firstly update the package by following command.

sudo apt update -y

```
ubuntu@ip-172-31-42-255:~$ sudo apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [382 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [83.9 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4704 B]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [277 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [117 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [10.4 kB]
Get:13 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.9 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:15 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2808 B]
Get:16 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get:17 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
```

```
Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [532 B]
Get:35 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:36 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:37 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.6 kB]
Get:38 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.8 kB]
Get:39 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [17.6 kB]
Get:40 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]
Get:41 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:42 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:43 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:44 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Fetched 28.2 MB in 6s (4883 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
6 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

14. Now,Install the required packages by executing following commands.

sudo apt install gcc -y

```
ubuntu@ip-172-31-42-255:~$ sudo apt install gcc -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu fontconfig-config
  fonts-dejavu-core fonts-dejavu-mono gcc-13 gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libaom3 libasan8 libatomic1
  libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libde265-0 libdeflate0
  libfontconfig1 libgcc-13-dev libgd3 libgomp1 libgprofng0 libheif-plugin-aomdec libheif-plugin-aomenc libheif-plugin-libde265
  libheif1 libhwasan0 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8 liblerc4 liblsan0 libmpc3 libquadmath0 libsframe1
  libsharpyuv0 libtiff6 libtsan2 libubsan1 libwebp7 libxpm4 linux-libc-dev manpages-dev rpcsvc-proto
Suggested packages:
  binutils-doc gprofng-gui cpp-doc gcc-13-locales cpp-13-doc gcc-multilib make autoconf automake libtool flex bison gdb gcc-doc
  gcc-13-multilib gcc-13-doc gdb-x86-64-linux-gnu glibc-doc libgd-tools libheif-plugin-x265 libheif-plugin-ffmpegdec
  libheif-plugin-jpegdec libheif-plugin-jpegenc libheif-plugin-j2kdec libheif-plugin-j2kenc libheif-plugin-rav1e
  libheif-plugin-svtenc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu fontconfig-config
  fonts-dejavu-core fonts-dejavu-mono gcc gcc-13 gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libaom3 libasan8
  libatomic1 libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libde265-0 libdeflate0
  libfontconfig1 libgcc-13-dev libgd3 libgomp1 libgprofng0 libheif-plugin-aomdec libheif-plugin-aomenc libheif-plugin-libde265
  libheif1 libhwasan0 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8 liblerc4 liblsan0 libmpc3 libquadmath0 libsframe1
  libsharpyuv0 libtiff6 libtsan2 libubsan1 libwebp7 libxpm4 linux-libc-dev manpages-dev rpcsvc-proto
0 upgraded, 57 newly installed, 0 to remove and 6 not upgraded.
Need to get 62.8 MB of archives.
After this operation, 222 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 binutils-common amd64 2.42-4ubuntu2 [239 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libsframe1 amd64 2.42-4ubuntu2 [14.8 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libbinutils amd64 2.42-4ubuntu2 [572 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libctf-nobfd0 amd64 2.42-4ubuntu2 [97.1 kB]
Setting up gcc (4:13.2.0-7ubuntu1) ...
Setting up libheif-plugin-aomdec:amd64 (1.17.6-1ubuntu4) ...
Setting up libheif1:amd64 (1.17.6-1ubuntu4) ...
Setting up libheif-plugin-libde265:amd64 (1.17.6-1ubuntu4) ...
Setting up libheif-plugin-aomenc:amd64 (1.17.6-1ubuntu4) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for sgml-base (1.31) ...
Setting up libfontconfig1:amd64 (2.15.0-1.1ubuntu2) ...
Setting up libgd3:amd64 (2.3.3-9ubuntu5) ...
Setting up libc-devtools (2.39-0ubuntu8.3) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-42-255:~$
```

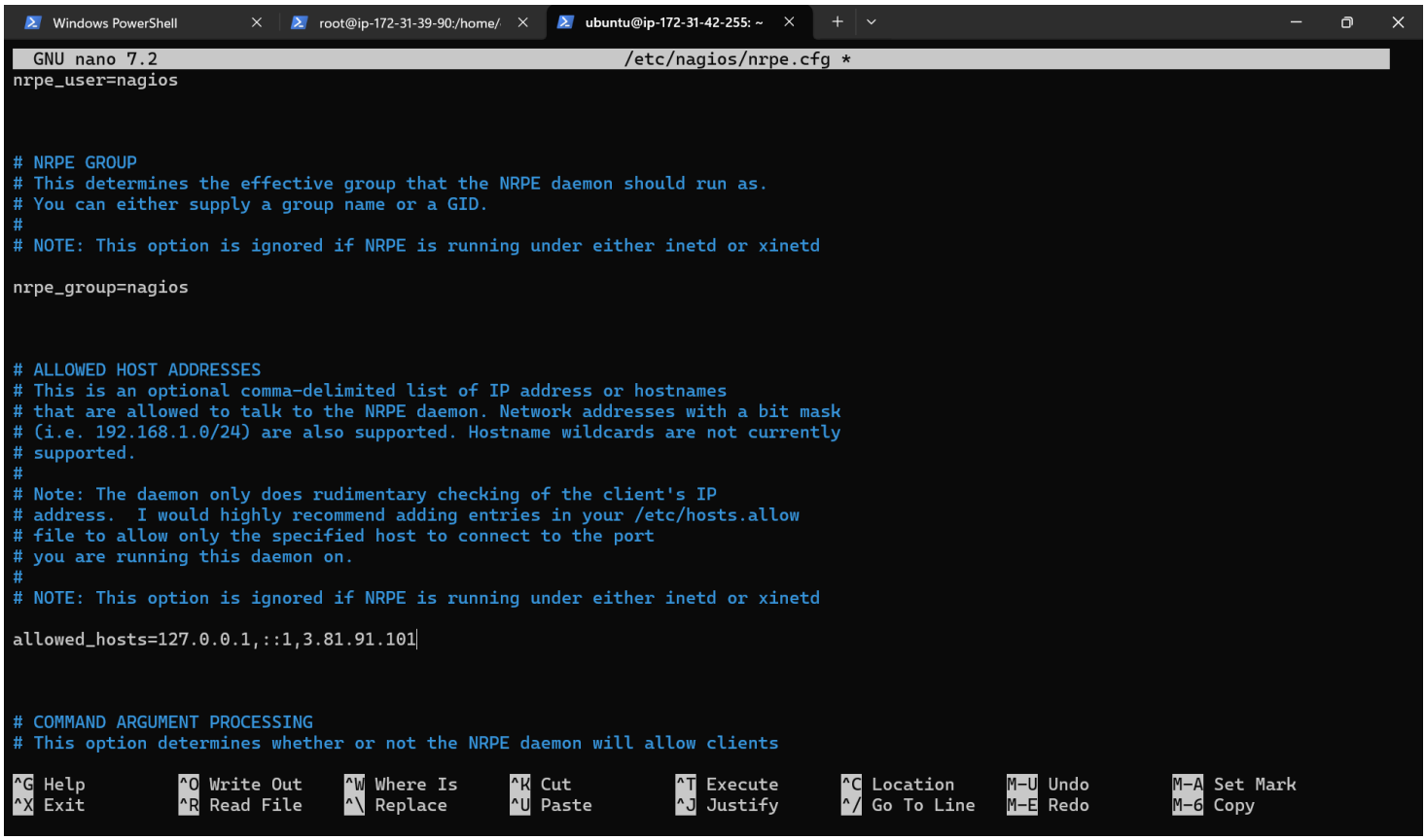
sudo apt install -y nagios-nrpe-server nagios-plugins

```
ubuntu@ip-172-31-42-255:~$ sudo apt install -y nagios-nrpe-server nagios-plugins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'
The following additional packages will be installed:
  libavahi-client3 libavahi-common-data libavahi-common3 libcups2t64 libdbi1t64 libldb2 libmysqlclient21 libnet-snmp-perl libpq5
  libradcli4 libsmbclient0 libsnmp-base libsnmp40t64 libtalloc2 libtdb1 libtevent0t64 liburiparser1 libwbclient0
  monitoring-plugins-basic monitoring-plugins-common monitoring-plugins-standard mysql-common python3-gpg python3-ldb
  python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common samba-common-bin samba-dsdb-modules samba-libs
  smbclient snmp
Suggested packages:
  cups-common libcrypt-des-perl libdigest-hmac-perl libio-socket-inet6-perl snmp-mibs-downloader icinga2 nagios-plugins-contrib
  fping postfix | sendmail-bin | exim4-daemon-heavy | exim4-daemon-light qstat xinetd | inetd python-markdown-doc heimdal-clients
  python3-dnspython cifs-utils
The following NEW packages will be installed:
  libavahi-client3 libavahi-common-data libavahi-common3 libcups2t64 libdbi1t64 libldb2 libmysqlclient21 libnet-snmp-perl libpq5
  libradcli4 libsmbclient0 libsnmp-base libsnmp40t64 libtalloc2 libtdb1 libtevent0t64 liburiparser1 libwbclient0 monitoring-plugins
  monitoring-plugins-basic monitoring-plugins-common monitoring-plugins-standard mysql-common nagios-nrpe-server python3-gpg
  python3-ldb python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common samba-common-bin samba-dsdb-modules
  samba-libs smbclient snmp
0 upgraded, 37 newly installed, 0 to remove and 6 not upgraded.
Need to get 16.1 MB of archives.
After this operation, 72.0 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 nagios-nrpe-server amd64 4.1.0-1ubuntu3 [356 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 rpcbind amd64 1.2.6-7ubuntu2 [46.5 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahi-common-data amd64 0.8-13ubuntu6 [29.7 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahi-common3 amd64 0.8-13ubuntu6 [23.3 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libavahi-client3 amd64 0.8-13ubuntu6 [26.8 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libcups2t64 amd64 2.4.7-1.2ubuntu7.3 [272 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libdbi1t64 amd64 0.9.0-6.1build1 [25.7 kB]
```

```
Creating config file /etc/nagios-plugins/config/netware.cfg with new version

Creating config file /etc/nagios-plugins/config/nt.cfg with new version

Creating config file /etc/nagios-plugins/config/pgsql.cfg with new version

Creating config file /etc/nagios-plugins/config/radius.cfg with new version

Creating config file /etc/nagios-plugins/config/rpc-nfs.cfg with new version

Creating config file /etc/nagios-plugins/config/snmp.cfg with new version
Setting up monitoring-plugins (2.3.5-1ubuntu3) ...
Setting up libldb2:amd64 (2:2.8.0+samba4.19.5+dfsg-4ubuntu9) ...
Setting up libavahi-client3:amd64 (0.8-13ubuntu6) ...
Setting up samba-libs:amd64 (2:4.19.5+dfsg-4ubuntu9) ...
Setting up python3-ldb (2:2.8.0+samba4.19.5+dfsg-4ubuntu9) ...
Setting up samba-dsdb-modules:amd64 (2:4.19.5+dfsg-4ubuntu9) ...
Setting up libsmbclient0:amd64 (2:4.19.5+dfsg-4ubuntu9) ...
Setting up libcups2t64:amd64 (2.4.7-1.2ubuntu7.3) ...
Setting up python3-samba (2:4.19.5+dfsg-4ubuntu9) ...
Setting up smbclient (2:4.19.5+dfsg-4ubuntu9) ...
Setting up samba-common-bin (2:4.19.5+dfsg-4ubuntu9) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-42-255:~$
```

15. Now open the nrpe.cfg file. Scroll down the page and find allowed_hosts.Go there, and add your nagios host IP address.

sudo nano /etc/nagios/nrpe.cfg



16. Now restart the NRPE server

sudo systemctl restart nagios-nrpe-server

17. Now again check the nagios-host status

sudo systemctl status nagios

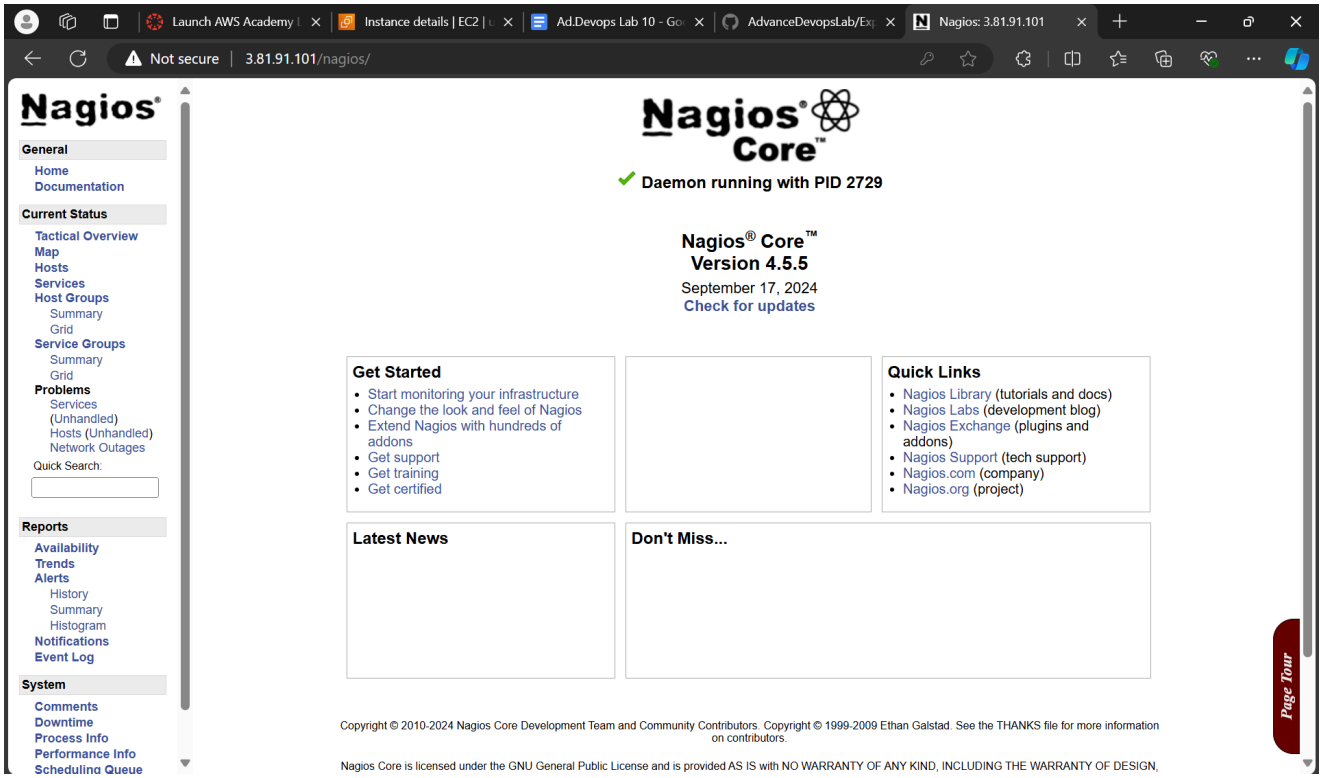
Now check the status is active or not

```
[root@ip-172-31-39-90 ec2-user]#
sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; disabled; preset: disabled)
     Active: active (running) since Fri 2024-10-04 09:24:28 UTC; 9min ago
       Docs: https://www.nagios.org/documentation
    Process: 2725 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCC>
    Process: 2727 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 2729 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 4.2M
        CPU: 114ms
     CGroup: /system.slice/nagios.service
             ├─2729 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─2730 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─2731 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─2732 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─2733 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─2742 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 09:30:28 ip-172-31-39-90.ec2.internal nagios[2729]: HOST ALERT: linuxserver;DOWN;SOFT;7;(No output on stdout) stderr>
Oct 04 09:31:28 ip-172-31-39-90.ec2.internal nagios[2729]: HOST ALERT: linuxserver;DOWN;SOFT;8;(No output on stdout) stderr>
Oct 04 09:32:28 ip-172-31-39-90.ec2.internal nagios[2729]: HOST ALERT: linuxserver;DOWN;SOFT;9;(No output on stdout) stderr>
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: HOST NOTIFICATION: nagiosadmin;linuxserver;DOWN;notify-host-by-e>
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: HOST ALERT: linuxserver;DOWN;HARD;10;(No output on stdout) stder>
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: wproc: NOTIFY job 6 from worker Core Worker 2732 is a non-check >
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: wproc:   host=linuxserver; service=(none); contact=nagiosadmin
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: wproc:   early_timeout=0; exited_ok=1; wait_status=32512; error_>
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: wproc:   stderr line 01: /bin/sh: line 1: /bin/mail: No such fil>
Oct 04 09:33:28 ip-172-31-39-90.ec2.internal nagios[2729]: wproc:   stderr line 02: /usr/bin/printf: write error: Broken pi>
[root@ip-172-31-39-90 ec2-user]#
```

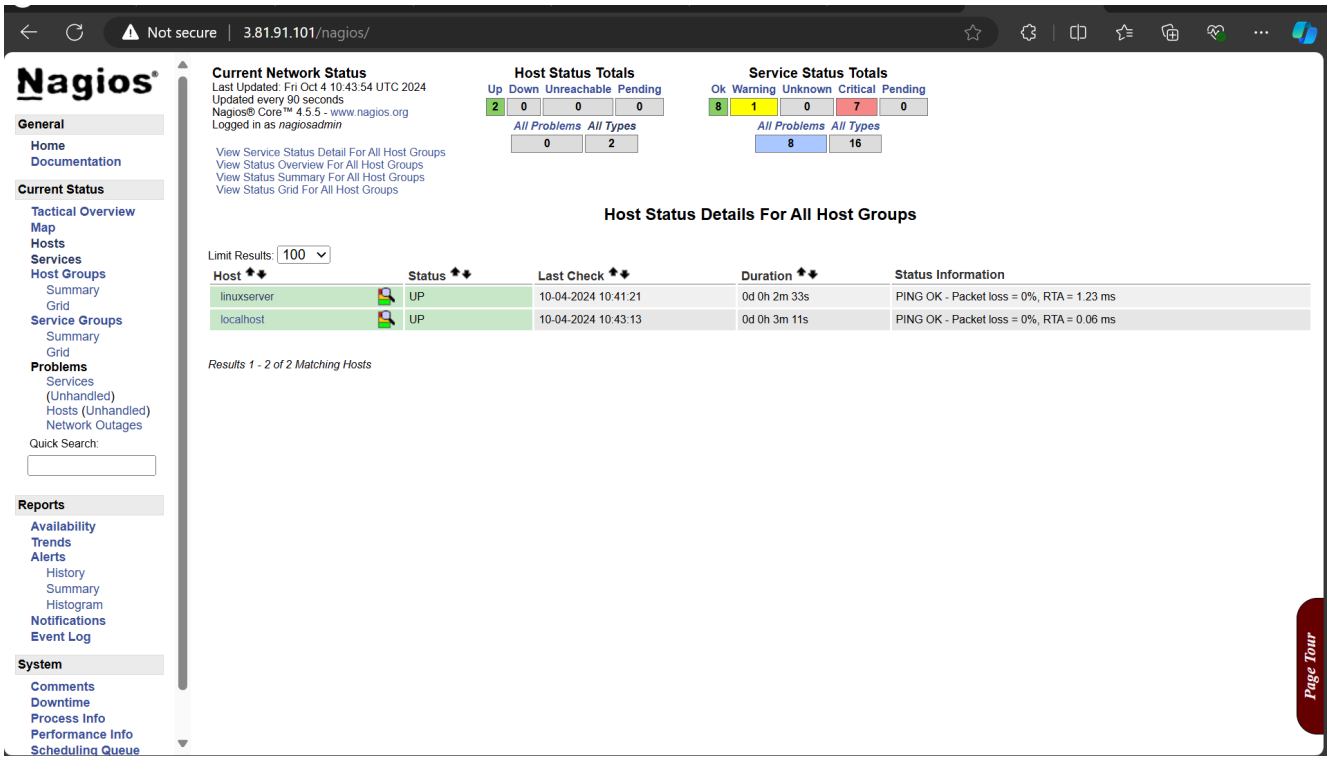
18. Now,check the status of http
    sudo systemctl status httpd
    sudo systemctl start httpd
    sudo systemctl enable httpd

```
[root@ip-172-31-39-90 ec2-user]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/httpd.service.d
             └─php-fpm.conf
     Active: active (running) since Fri 2024-10-04 09:45:14 UTC; 33min ago
       Docs: man:httpd.service(8)
   Main PID: 4146 (httpd)
     Status: "Total requests: 19; Idle/Busy workers 100/0;Requests/sec: 0.0095; Bytes served/sec:  70 B/sec"
      Tasks: 230 (limit: 1112)
     Memory: 17.3M
        CPU: 1.428s
     CGroup: /system.slice/httpd.service
             ├─4146 /usr/sbin/httpd -DFOREGROUND
             ├─4148 /usr/sbin/httpd -DFOREGROUND
             ├─4149 /usr/sbin/httpd -DFOREGROUND
             ├─4150 /usr/sbin/httpd -DFOREGROUND
             ├─4151 /usr/sbin/httpd -DFOREGROUND
             └─4533 /usr/sbin/httpd -DFOREGROUND

Oct 04 09:45:14 ip-172-31-39-90.ec2.internal systemd[1]: Stopped httpd.service - The Apache HTTP Server.
Oct 04 09:45:14 ip-172-31-39-90.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Oct 04 09:45:14 ip-172-31-39-90.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Oct 04 09:45:14 ip-172-31-39-90.ec2.internal httpd[4146]: Server configured, listening on: port 80
[root@ip-172-31-39-90 ec2-user]# sudo systemctl start httpd
[root@ip-172-31-39-90 ec2-user]# sudo systemctl enable httpd
```

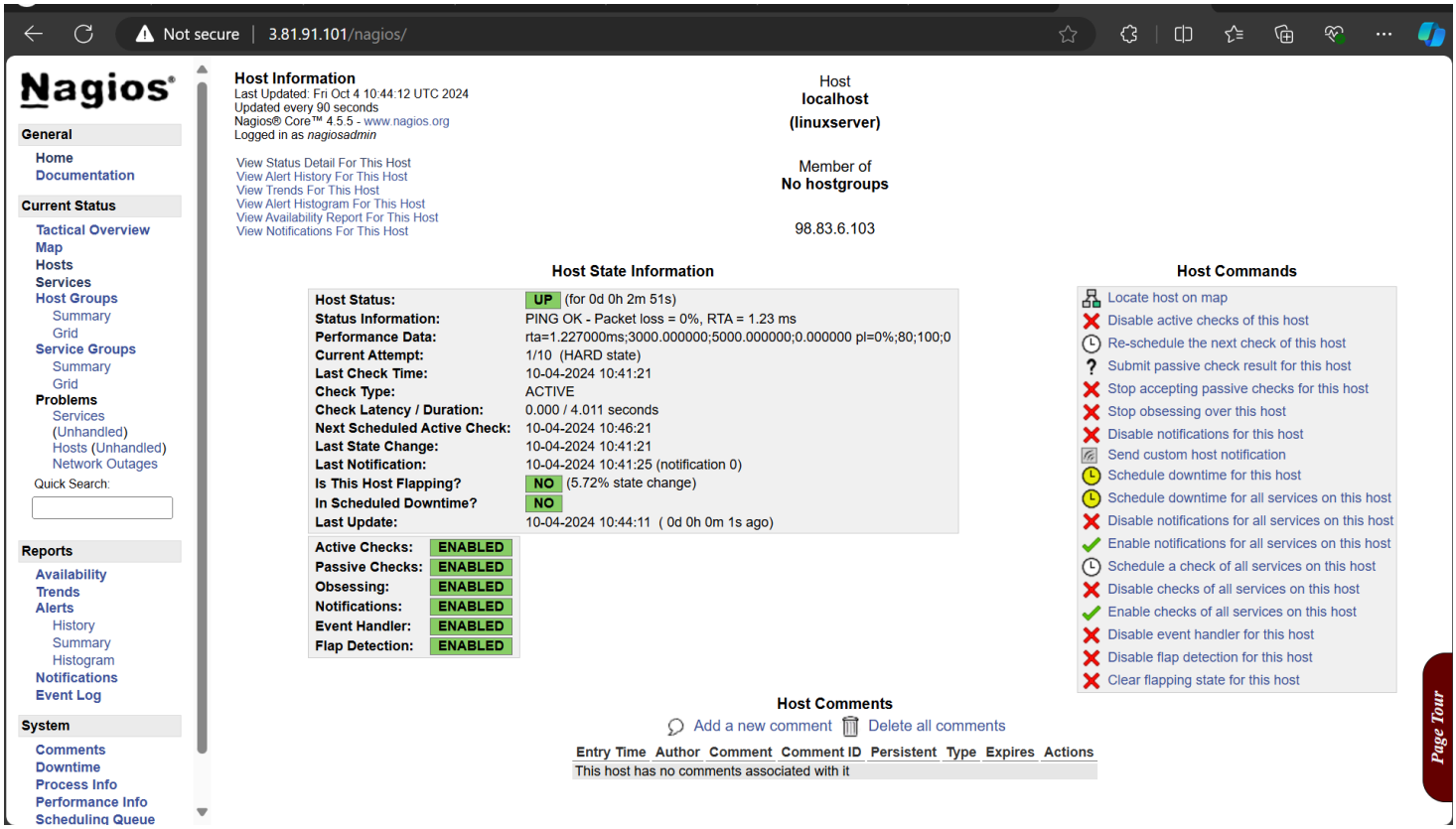
19. Now to check Nagios dashboard go to http://<Nagios-host ip>/nagios.
    Take host public IP address to add in “Nagios-host ip”.



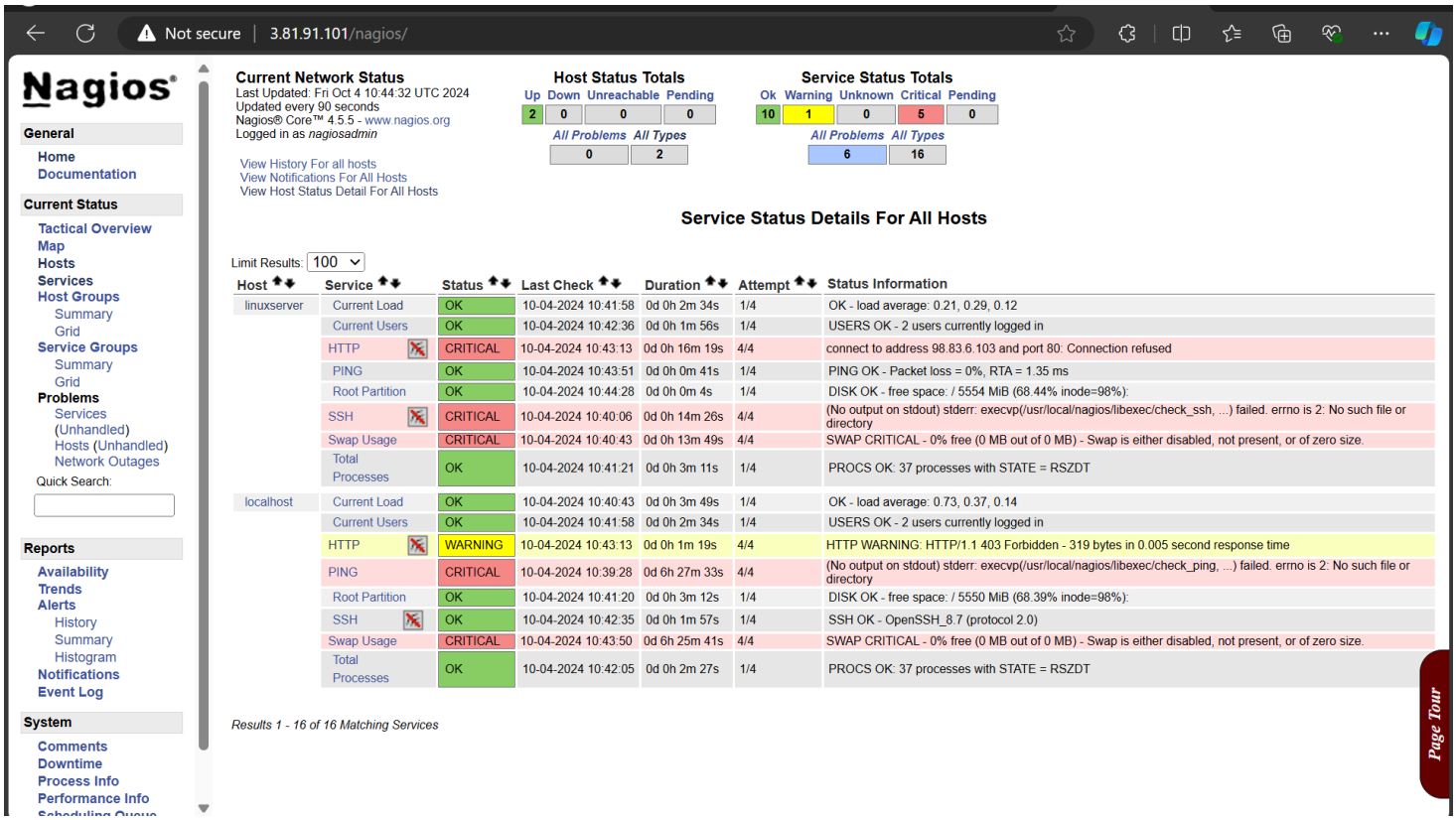
20. Now Click on Hosts from the left side panel there you will get the status of linuxserver and localhost. Check whether its status is up or down.If it is up then it is successful.



21. Now click on linuxserver. There you will get Host state information.



22. Then click on Services presnet on left sidebar where you will get current network status.



Conclusion: In this experiment, we created a nagios client EC-2 instance of t2.micro ubuntu instance. Then after updating hostname, hostgroup_name.Then adding the address to the public IP of your Linux client.Then verify the configuration and then check the status of nagios-host and then start and enable it then open http://<Nagios-host ip>/nagios. There you will get the status of linuxserver and localhost.Thus we can monitor essential network services on the Linux server.