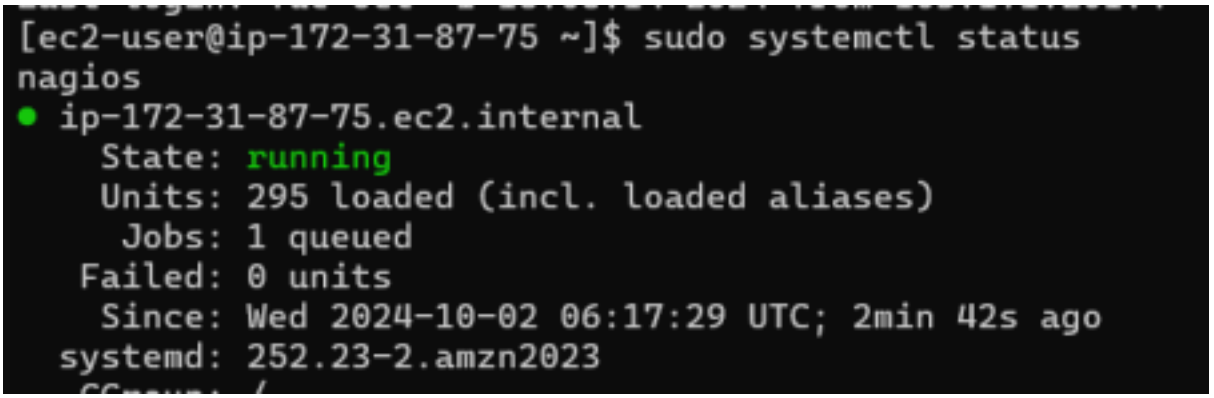
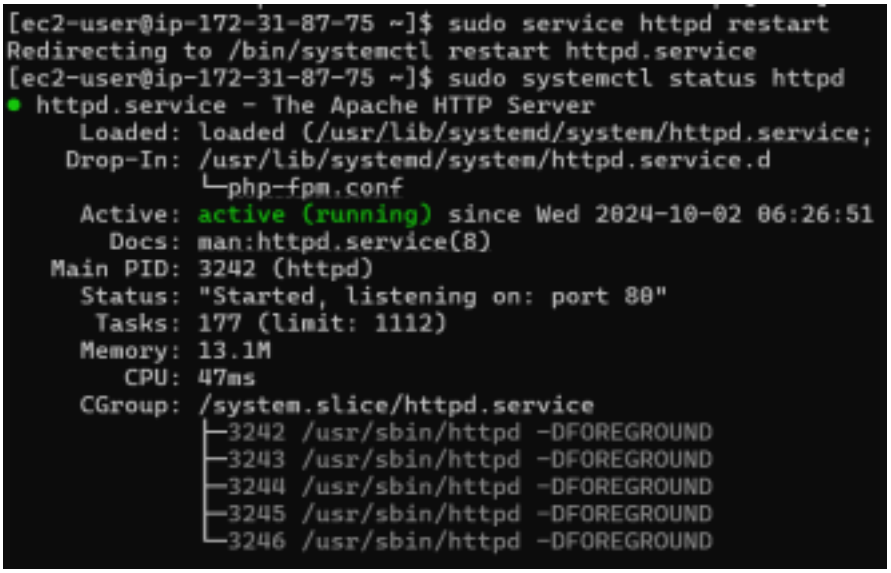
Experiment 10

**Aim: To perform Port, Service monitoring, Windows/Linux server monitoring using**

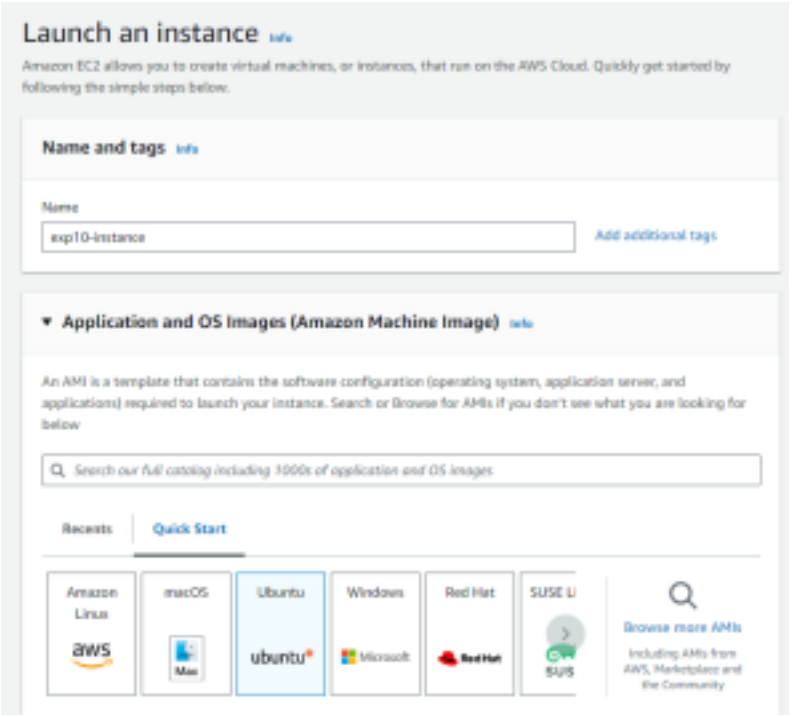
**Nagios. Steps:**

1. Firstly, we will check whether nagios is running on the server side by using the command “sudo systemctl status nagios” on the host machine (host machine is the instance connected to the terminal in experiment 9, ensure that you have started the instance created for exp9, also check status of apache).

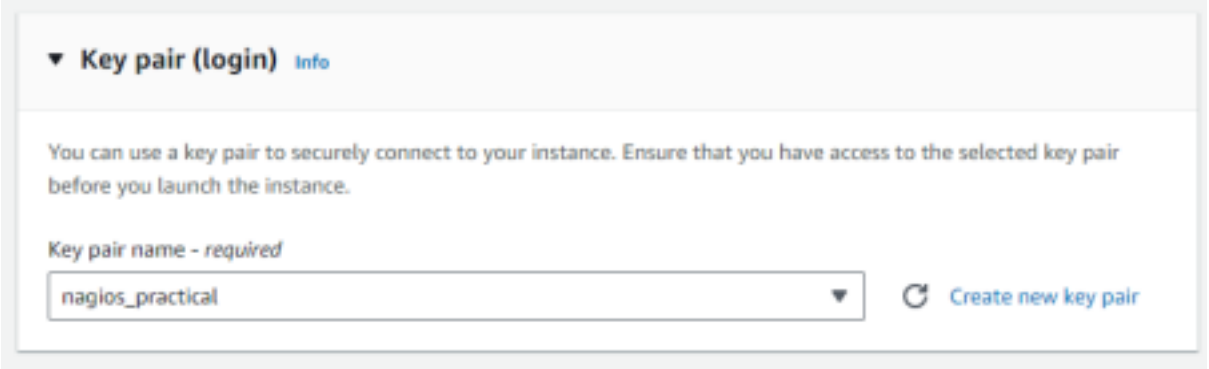


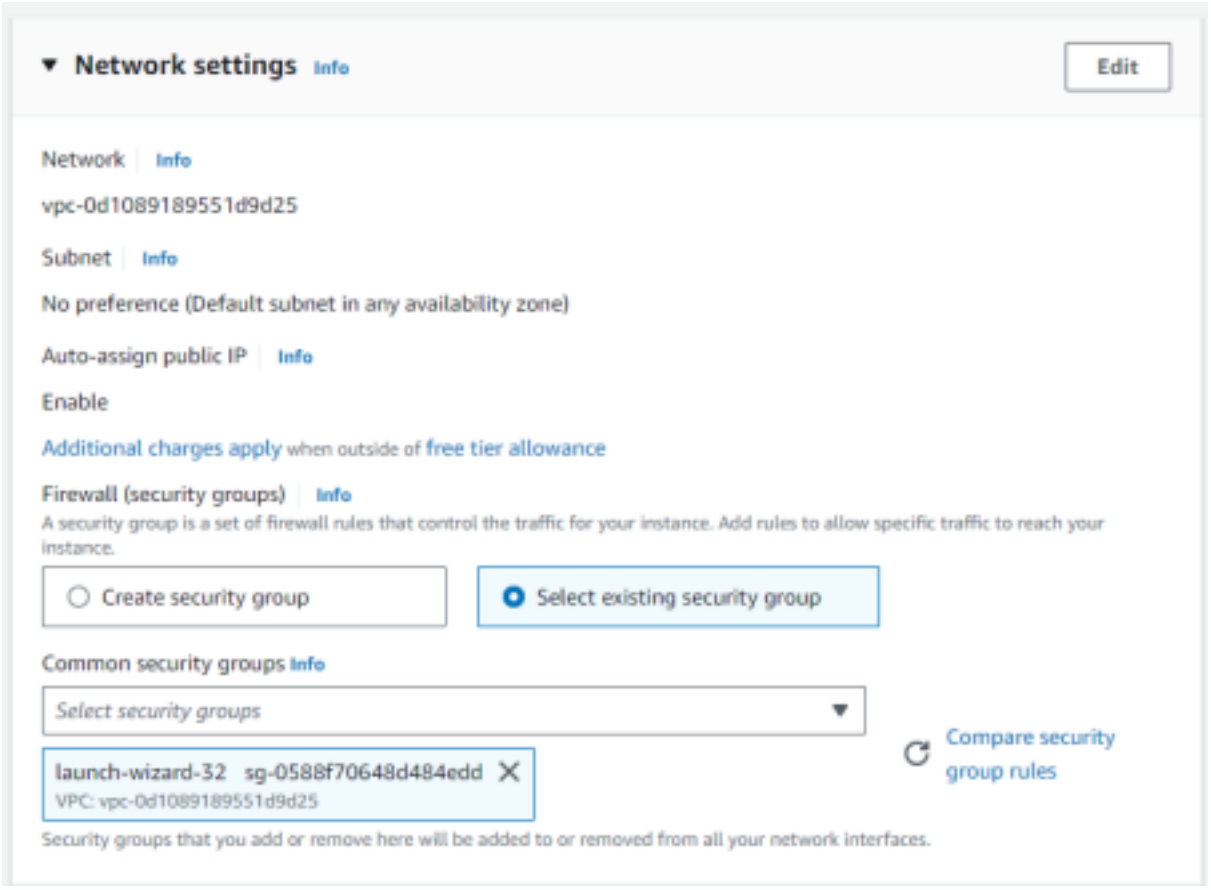

2. Now we will launch a new instance. Select ubuntu for the OS.



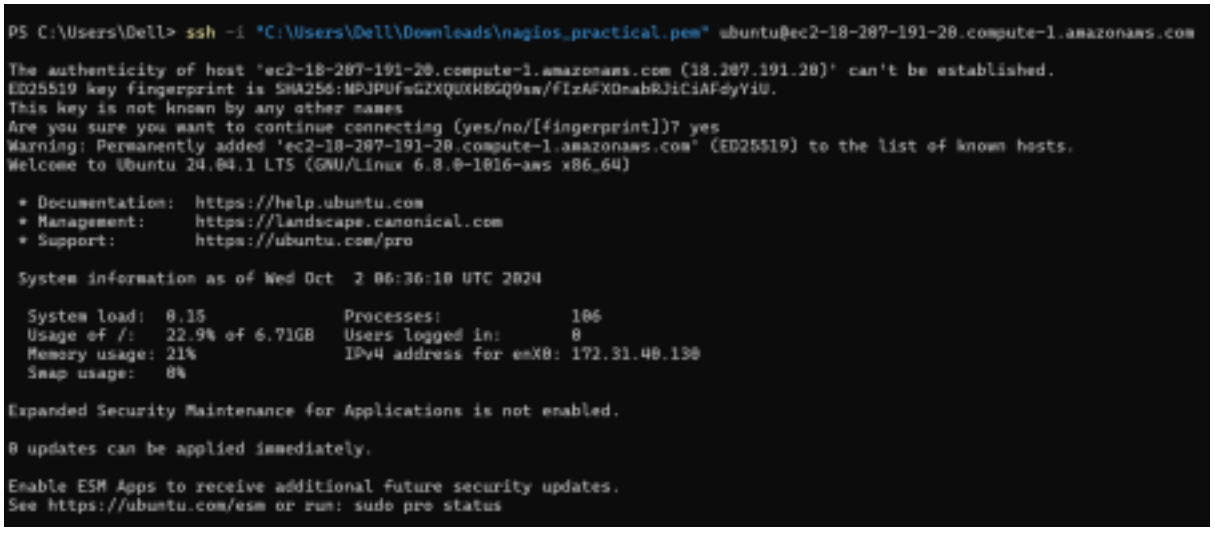

3. Select the key pair which was created and used in the exp 9.




4. Select existing security group and from the list of options select the security group created for exp 9. Previously it was launch wizard 32 and so here I have selected the same.




5. Open a new terminal to connect to the client machine. Copy the SSH command provided in the SSH client section during connection of instance. When pasting the command into your terminal, ensure you specify the full path to your .pem file instead of just the file name.




6. Now go back to your host machine and run the following command

ps -ef | grep nagios

```
[ec2-user@ip-172-31-87-75 ~]$ ps -ef | grep nagios
nagios      2002       1  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios      2003    2002  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2004    2002  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2005    2002  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2006    2002  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2007    2002  0 06:17 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user    4389    2306  0 06:43 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-87-75 ~]$
```

7. Now perform these commands on the host terminal
sudo su
mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts

```
[root@ip-172-31-87-75 ec2-user]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-87-75 ec2-user]#
```

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

```
[root@ip-172-31-87-75 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linux
[root@ip-172-31-87-75 ec2-user]#
```

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

The above given command will open the nano text editor wherein you have to do the following changes:

i. Change the hostgroup name to linux-servers1

```
###############################################################################
#
# HOST GROUP DEFINITION
#
###############################################################################

# Define an optional hostgroup for Linux machines

define hostgroup {

    hostgroup_name          linux-servers1          ; The name of the hostgroup
    alias                   Linux Servers           ; Long name of the group
    members                 linuxserver             ; Comma separated list of hosts that belong to this group
}
```

ii. Change host name and alias from localhost to linuxserver everywhere in the file

```
# Define a service to "ping" the local machine

define service {

    use                     local-service           ; Name of service template to use
    host_name               linuxserver
    service_description     PING
    check_command           check_ping!100.0,20%!500.0,60%
}
```

iii. Change the address to the public IPv4 address of the ubuntu instance (You will find the ip address when you select the instance on the ec2 instances dashboard)

```
# Define a host for the local machine

define host {

    use                     linux-server            ; Name of host template to use
                                                    ; This host definition will inherit all variables that are defined
                                                    ; in (or inherited by) the linux-server host template definition.
    host_name               linuxserver
    alias                   linuxserver
    address                 18.207.191.20
}
```

8. Open the Nagios Config file by using this command:
   nano /usr/local/nagios/etc/nagios.cfg
   nano text editor will get opened

```
##############################################################################
#
# NAGIOS.CFG - Sample Main Config File for Nagios 4.5.5
#
# Read the documentation for more information on this configuration
# file.  I've provided some comments here, but things may not be so
# clear without further explanation.
#
#
##############################################################################


# LOG FILE
# This is the main log file where service and host events are logged
# for historical purposes.  This should be the first option specified
# in the config file!!!

log_file=/usr/local/nagios/var/nagios.log



# OBJECT CONFIGURATION FILE(S)
# These are the object configuration files in which you define hosts,
# host groups, contacts, contact groups, services, etc.
# You can split your object definitions across several config files
# if you wish (as shown below), or keep them all in a single config file.

# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
```

9. In the text editor add “cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/” this line

```
# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:

#cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
```

10. Now we will verify the configuration files

    /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 16 services.
        Checked 2 hosts.
        Checked 2 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 2 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
[root@ip-172-31-87-75 ec2-user]#
```

If there are no errors we can proceed further

11. We will now restart the nagios service
    service nagios restart

```
[root@ip-172-31-87-75 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-87-75 ec2-user]#
```

12. Now on the client machine (The ubuntu machine we created for this experiment) run the following command:

sudo apt update -y

```
ubuntu@ip-172-31-40-130:~$ 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://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [380 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [535 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [130 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [8676 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [380 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [156 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [45.0 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [14.9 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.4 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3608 B]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [212 B]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [532 B]
```

sudo apt install gcc -y

```
ubuntu@ip-172-31-40-130:~$ 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.
```

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

```
ubuntu@ip-172-31-40-130:~$ 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]
```

13. Open nrpe.cfg file to make changes.
    sudo nano /etc/nagios/nrpe.cfg
    Under allowed_hosts, add your nagios host public IPv4 address:

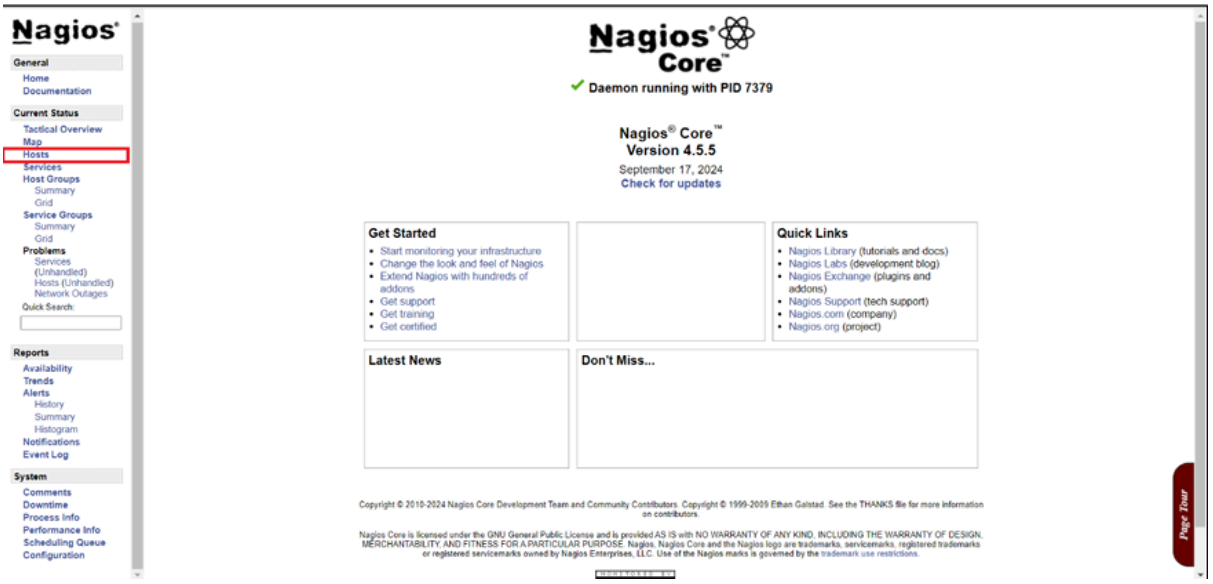
```
#
# Note: The daemon only does rudimentary checking of the client's IP
# address.  I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

allowed_hosts=127.0.0.1,54.163.184.143


# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments to commands that are executed.  This option only works
# if the daemon was configured with the --enable-command-args configure script
# option.
#
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
# Read the SECURITY file for information on some of the security implications
# of enabling this variable.
#
# Values: 0=do not allow arguments, 1=allow command arguments
```

14. Now restart the NRPE server

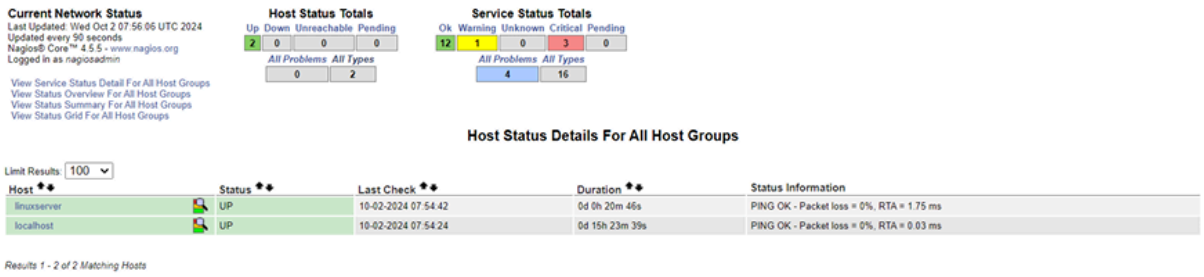
    sudo systemctl restart nagios-nrpe-server

```
ubuntu@ip-172-31-40-130:~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-40-130:~$
```

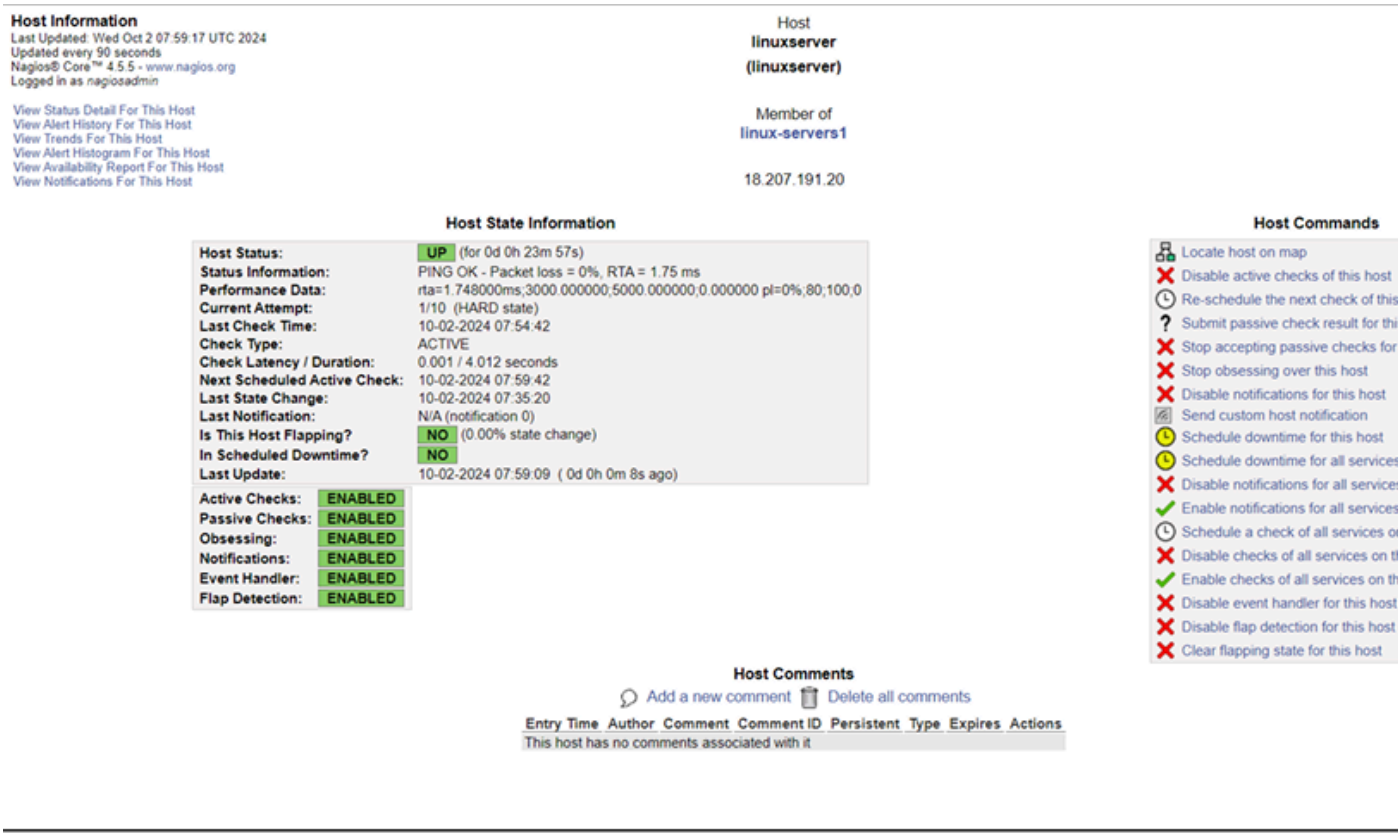
15. Go to the nagios dashboard and click on hosts



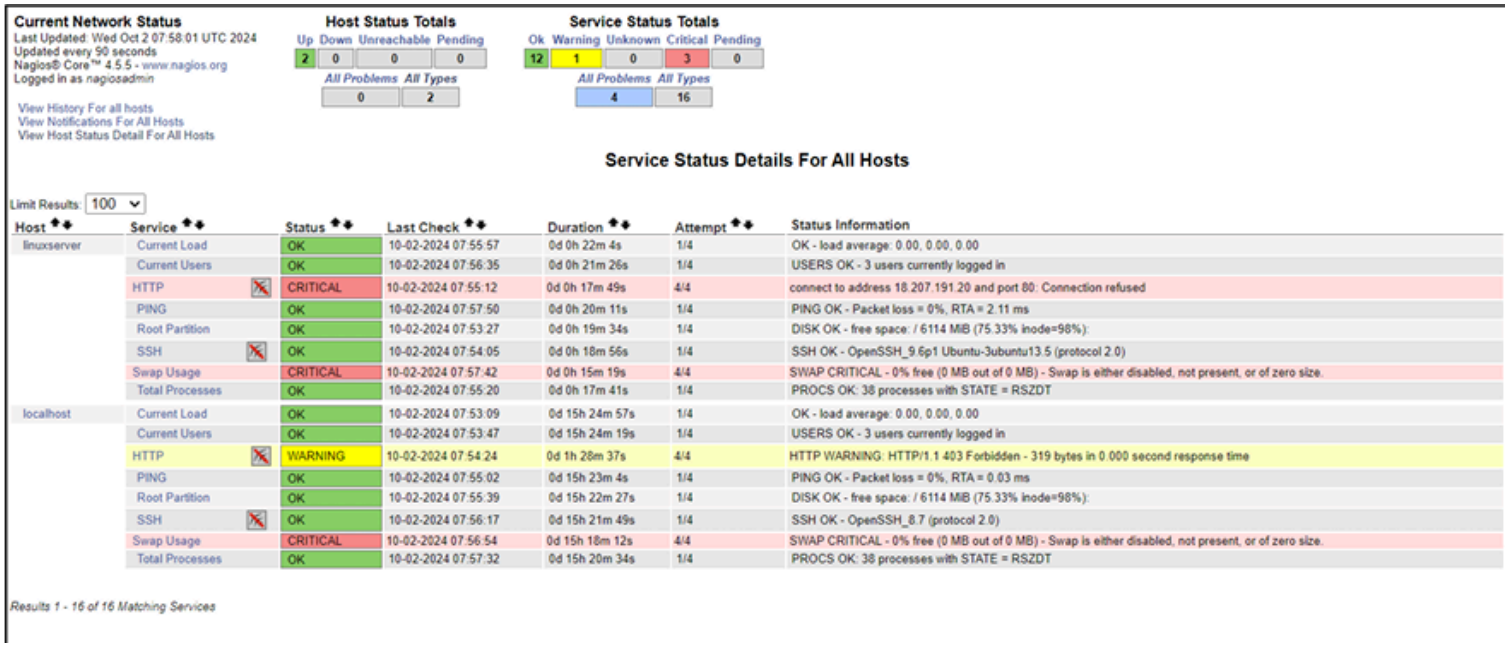

Click on linux server




We can see the host state information:




If you want to see all the services and ports being monitored then select the services option and you will see the page as shown below:



**Current Network Status**
Last Updated: Wed Oct 2 07:58:01 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

**Host Status Totals**

| Up | Down | Unreachable | Pending |
|---|---|---|---|
| 2 | 0 | 0 | 0 |

| All Problems | All Types |
|---|---|
| 0 | 2 |

**Service Status Totals**

| Ok | Warning | Unknown | Critical | Pending |
|---|---|---|---|---|
| 12 | 1 | 0 | 3 | 0 |

| All Problems | All Types |
|---|---|
| 4 | 16 |

**Service Status Details For All Hosts**

Limit Results: 100

| Host | Service | Status | Last Check | Duration | Attempt | Status Information |
|---|---|---|---|---|---|---|
| linuxserver | Current Load | OK | 10-02-2024 07:55:57 | 0d 0h 22m 4s | 1/4 | OK - load average: 0.00, 0.00, 0.00 |
| | Current Users | OK | 10-02-2024 07:56:35 | 0d 0h 21m 26s | 1/4 | USERS OK - 3 users currently logged in |
| | HTTP | CRITICAL | 10-02-2024 07:55:12 | 0d 0h 17m 49s | 4/4 | connect to address 18.207.191.20 and port 80: Connection refused |
| | PING | OK | 10-02-2024 07:57:50 | 0d 0h 20m 11s | 1/4 | PING OK - Packet loss = 0%, RTA = 2.11 ms |
| | Root Partition | OK | 10-02-2024 07:53:27 | 0d 0h 19m 34s | 1/4 | DISK OK - free space: / 6114 MiB (75.33% inode=98%): |
| | SSH | OK | 10-02-2024 07:54:05 | 0d 0h 18m 56s | 1/4 | SSH OK - OpenSSH_9.6p1 Ubuntu-3ubuntu13.5 (protocol 2.0) |
| | Swap Usage | CRITICAL | 10-02-2024 07:57:42 | 0d 0h 15m 19s | 4/4 | SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size. |
| | Total Processes | OK | 10-02-2024 07:55:20 | 0d 0h 17m 41s | 1/4 | PROCS OK: 38 processes with STATE = RSZDT |
| localhost | Current Load | OK | 10-02-2024 07:53:09 | 0d 15h 24m 57s | 1/4 | OK - load average: 0.00, 0.00, 0.00 |
| | Current Users | OK | 10-02-2024 07:53:47 | 0d 15h 24m 19s | 1/4 | USERS OK - 3 users currently logged in |
| | HTTP | WARNING | 10-02-2024 07:54:24 | 0d 1h 28m 37s | 4/4 | HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.000 second response time |
| | PING | OK | 10-02-2024 07:55:02 | 0d 15h 23m 4s | 1/4 | PING OK - Packet loss = 0%, RTA = 0.03 ms |
| | Root Partition | OK | 10-02-2024 07:55:39 | 0d 15h 22m 27s | 1/4 | DISK OK - free space: / 6114 MiB (75.33% inode=98%): |
| | SSH | OK | 10-02-2024 07:56:17 | 0d 15h 21m 49s | 1/4 | SSH OK - OpenSSH_8.7 (protocol 2.0) |
| | Swap Usage | CRITICAL | 10-02-2024 07:56:54 | 0d 15h 18m 12s | 4/4 | SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size. |
| | Total Processes | OK | 10-02-2024 07:57:32 | 0d 15h 20m 34s | 1/4 | PROCS OK: 38 processes with STATE = RSZDT |

Results 1 - 16 of 16 Matching Services

Conclusion: To conduct this experiment, it's necessary to start the instance from the previous experiment, as it will serve as the host, while the instance created in this experiment will act as the client machine. When I attempted to run the command to verify the Nagios configuration file, I encountered errors. To resolve these errors, I reinstalled the Nagios plugins and restarted the Nagios service, which fixed the issues.