Name:Shejal Tiwari Class/Roll No: D15C/ 57

Experiment No: 10

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

Theory:

Port and Service Monitoring
Port and service monitoring in Nagios involves checking the availability and responsiveness of network services running on specific ports. This ensures that critical services (like HTTP, FTP, or SSH) are operational. Nagios uses plugins to ping the ports and verify whether services are up and responding as expected, allowing administrators to be alerted in case of outages.

Windows/Linux Server Monitoring
Windows/Linux server monitoring with Nagios entails tracking the performance and health of servers running these operating systems. It includes monitoring metrics such as CPU usage, memory consumption, disk space, and system logs. Nagios employs various plugins to gather data, enabling administrators to ensure optimal performance, identify potential issues, and maintain uptime across their server infrastructure.

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

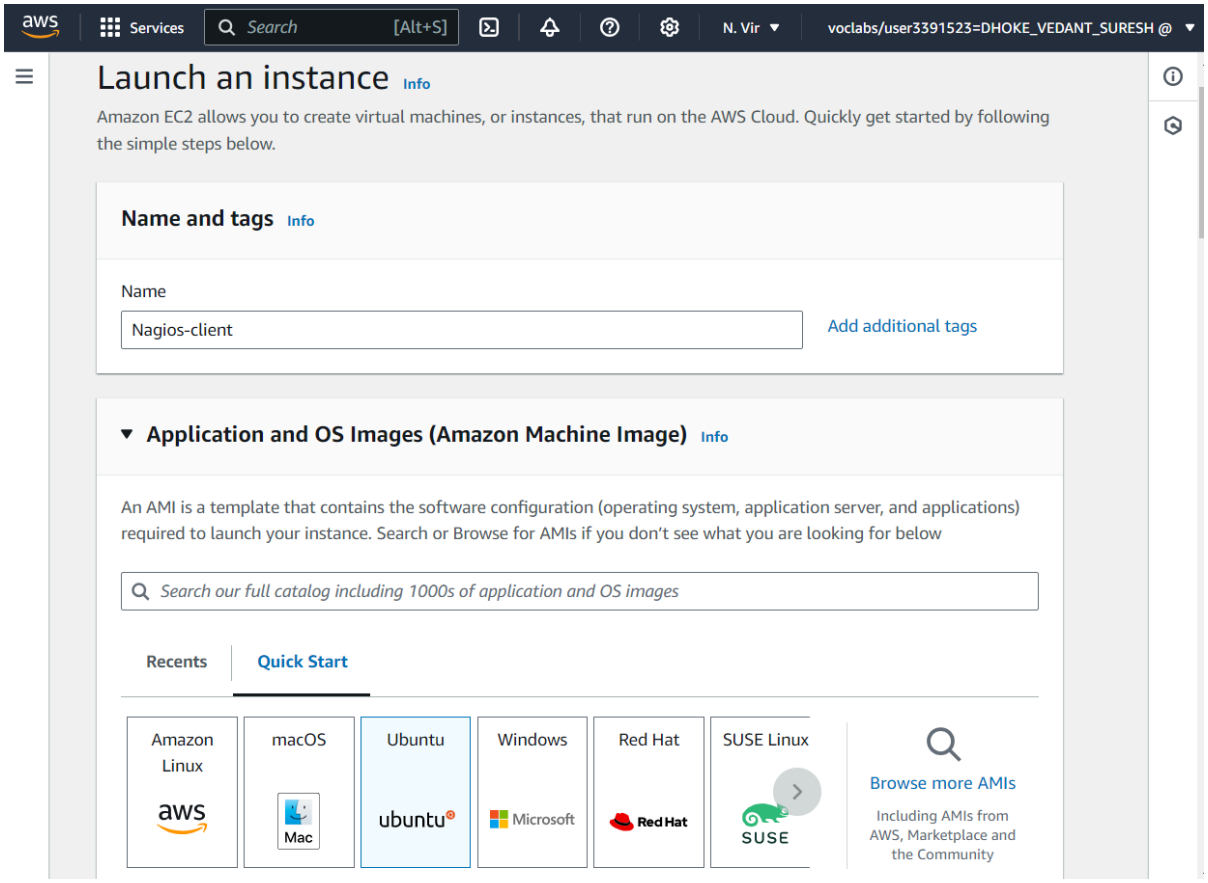
Monitoring Using Nagios:

Step 1: To Confirm Nagios is running on the server side Perform the following command on your Amazon Linux Machine (Nagios-host).
sudo systemctl status nagios

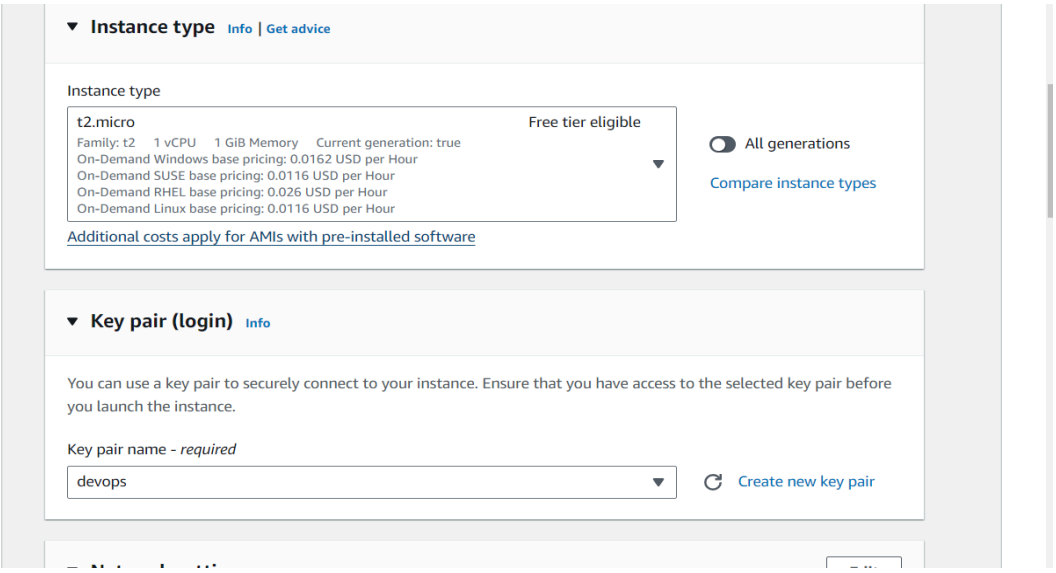
```
ec2-user@ip-172-31-46-196:~
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; disabled; preset: disabled)
     Active: active (running) since Sun 2024-10-06 10:58:43 UTC; 4s ago
       Docs: https://www.nagios.org/documentation
    Process: 62217 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
    Process: 62218 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 62219 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.4M
        CPU: 74ms
     CGroup: /system.slice/nagios.service
             ├─62219 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─62220 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─62221 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─62222 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─62223 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─62224 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: qh: core query handler registered
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: qh: echo service query handler registered
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: qh: help for the query handler registered
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: wproc: Successfully registered manager as @wproc with query handler
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: wproc: Registry request: name=Core Worker 62223;pid=62223
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: wproc: Registry request: name=Core Worker 62221;pid=62221
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: wproc: Registry request: name=Core Worker 62222;pid=62222
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: wproc: Registry request: name=Core Worker 62220;pid=62220
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: Successfully launched command file worker with pid 62224
Oct 06 10:58:43 ip-172-31-46-196.ec2.internal nagios[62219]: HOST ALERT: localhost;DOWN;SOFT;1;(No output on stdout) stderr: execvp(/usr/local/nagios/libex>
~
```

You can now proceed if you get the above message/output.

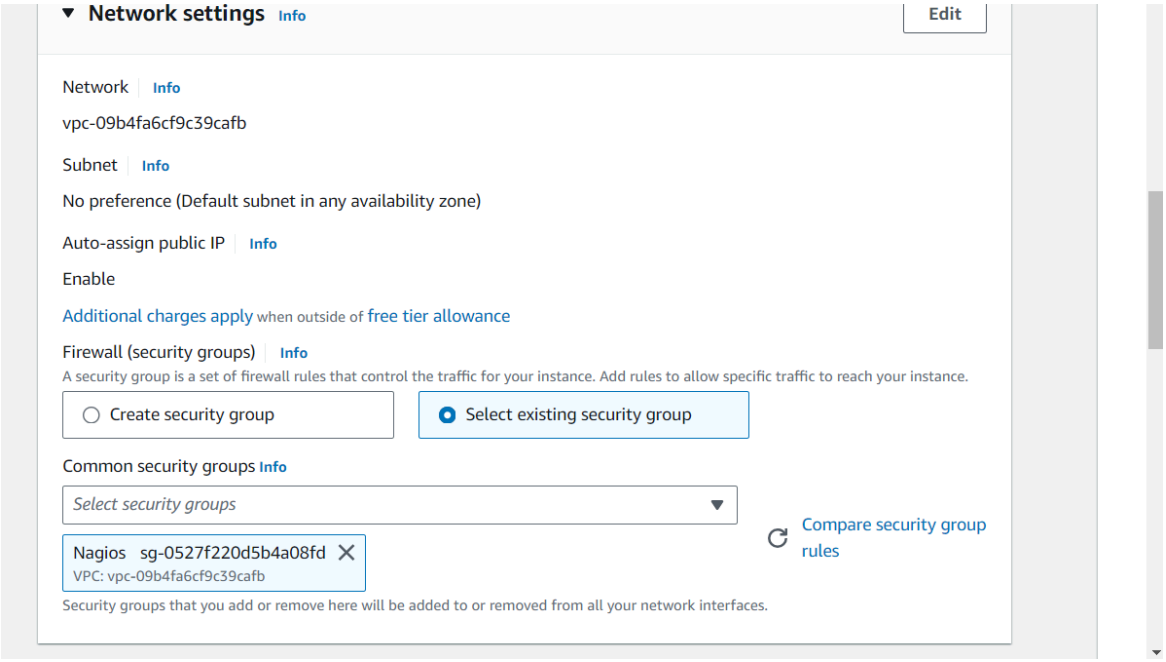
Step 2: Now Create a new EC2 instance. Name: Nagios-client, AMI: Ubuntu Instance Type: t2.micro.



For Key pair : Click on create key and make key of type RSA with extension .pem . Key will be downloaded to your local machine.



Select the Existing Security Group and select the Security Group that we have created in Experiment no 9 or the same one you have used for the Nagios server (Nagios-host).



Step 3: Now After creating the EC2 Instance click on connect and then copy the command which is given as example in the SSH Client section .

Now open the terminal in the folder where your key(RSA key with .pem) is located. and paste that copied command.Successfully connected to the instance.

```
PS C:\Users\Vedant> ssh -i "devops.pem" ubuntu@ec2-3-81-218-241.compute-1.amazonaws.com
The authenticity of host 'ec2-3-81-218-241.compute-1.amazonaws.com (3.81.218.241)' can't be establish
ed.
ED25519 key fingerprint is SHA256:7YtdUbwcFY6vK575h5DIfKqnlOf22OVC34blKsmOQcw.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-81-218-241.compute-1.amazonaws.com' (ED25519) to the list of known
hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Sep 27 08:38:26 UTC 2024

  System load:    1.36      Processes:             26
  Usage of /home: unknown   Users logged in:       0
  Memory usage:   4%        IPv4 address for eth0: 10.10.10.2
  Swap usage:     0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

Now perform all the commands on the Nagios-host till step 10

Step 4: Now on the server Nagios-host run the following command.

ps -ef | grep nagios

```
[ec2-user@ip-172-31-46-196 ~]$ ps -ef | grep nagios
nagios      2428       1  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagio
/etc/nagios.cfg
nagios      2430    2428  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local
nagios/var/rw/nagios.qh
nagios      2431    2428  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local
nagios/var/rw/nagios.qh
nagios      2432    2428  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local
nagios/var/rw/nagios.qh
nagios      2433    2428  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local
nagios/var/rw/nagios.qh
nagios      2437    2428  0 11:05 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagio
/etc/nagios.cfg
ec2-user    3367    3273  0 11:16 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-46-196 ~]$
```

Step 5: Now Become root user and create root directories.

sudo su

mkdir /usr/local/nagios/etc/objects/monitorhosts

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

```
[ec2-user@ip-172-31-46-196 ~]$ sudo su
[root@ip-172-31-46-196 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-46-196 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-46-196 ec2-user]#
```

6: Copy the sample localhost.cfg to linuxhost.cfg by running the following command.(Below command should come in one line see screenshot below)

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

```
[root@ip-172-31-46-196 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-46-196 ec2-user]#
```

Step 7: Open linuxserver.cfg using nano and make the following changes in all positions?everywhere in file.

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

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

```
root@ip-172-31-46-196:/home   ubuntu@ip-172-31-37-150: ~
  GNU nano 5.8     /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg      Modified


###############################################################################
#
# HOST DEFINITION
#
###############################################################################

# 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 t>
                                                    ; in (or inherited by) the linux-server host templa>
    host_name               linux-server
    alias                   localhost
    address                 3.81.218.241
}



###############################################################################
#
# 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                 localhost               ; Comma separated list of hosts that belong to this>
}



^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify    ^/ Go To Line M-E Redo
```

Step 8: Now update the Nagios config file .Add the following line in the file.

Line to add : cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

Run the command : **nano /usr/local/nagios/etc/nagios.cfg**

```
  GNU nano 5.8                    /usr/local/nagios/etc/nagios.cfg                    Modified
##############################################################################
#
# 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
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch

^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify    ^/ Go To Line M-E Redo
```

Step 9: Now Verify the configuration files by running the following commands.
/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```
[root@ip-172-31-46-196 ec2-user]# /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...
Warning: Duplicate definition found for service 'HTTP' on host 'localhost' (config file '/usr/local/nagios
fg', starting on line 152)
Warning: Duplicate definition found for service 'SSH' on host 'localhost' (config file '/usr/local/nagios/
g', starting on line 138)
Warning: Duplicate definition found for service 'Swap Usage' on host 'localhost' (config file '/usr/local/
rver.cfg', starting on line 125)
Warning: Duplicate definition found for service 'Current Load' on host 'localhost' (config file '/usr/loca
server.cfg', starting on line 112)
Warning: Duplicate definition found for service 'Total Processes' on host 'localhost' (config file '/usr/l
nuxserver.cfg', starting on line 100)
Warning: Duplicate definition found for service 'Current Users' on host 'localhost' (config file '/usr/loc
xserver.cfg', starting on line 86)
Warning: Duplicate definition found for service 'Root Partition' on host 'localhost' (config file '/usr/lo
uxserver.cfg', starting on line 72)
Warning: Duplicate definition found for service 'PING' on host 'localhost' (config file '/usr/local/nagios
fg', starting on line 58)
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 8 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

[root@ip-172-31-46-196 ec2-user]#
```

Step 10: Now restart the services of nagios by running the following command.

service nagios restart

```
[root@ip-172-31-46-196 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.servicetart
[root@ip-172-31-46-196 ec2-user]#
```

Step 11: Now Go to the Nagios-client ssh terminal and update and install the packages by running the following command.

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

```
ubuntu@ip-172-31-37-150:~$ sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
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://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [83.9 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4704 B]
Get:10 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [277 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [117 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:16 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [537 kB]
Get:20 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [10.4 kB]
Get:21 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.9 kB]
```

```
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-172-31-37-150:~$
```

Step 12: Open nrpe.cfg file to make changes.Under allowed_hosts, add your nagios host IP address.
sudo nano /etc/nagios/nrpe.cfg

```
# NRPE USER
# This determines the effective user that the NRPE daemon should run as.
# You can either supply a username or a UID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

nrpe_user=nagios



# NRPE GROUP
# This determines the effective group that the NRPE daemon should run as.
# You can either supply a group name or a GID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

nrpe_group=nagios



# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon. Network addresses with a bit mask
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
# supported.
#
# 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,::1,3.91.89.94



# COMMAND ARGUMENT PROCESSING

^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify    ^/ Go To Line M-E Redo
```

Step 13: Now restart the NRPE server by this command.
sudo systemctl restart nagios-nrpe-server

```
ubuntu@ip-172-31-37-150:~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-37-150:~$
```

Step 14: Now again check the status of Nagios by running this command on Nagios-host and also check httpd is active and run the command to active it.
sudo systemctl status nagios

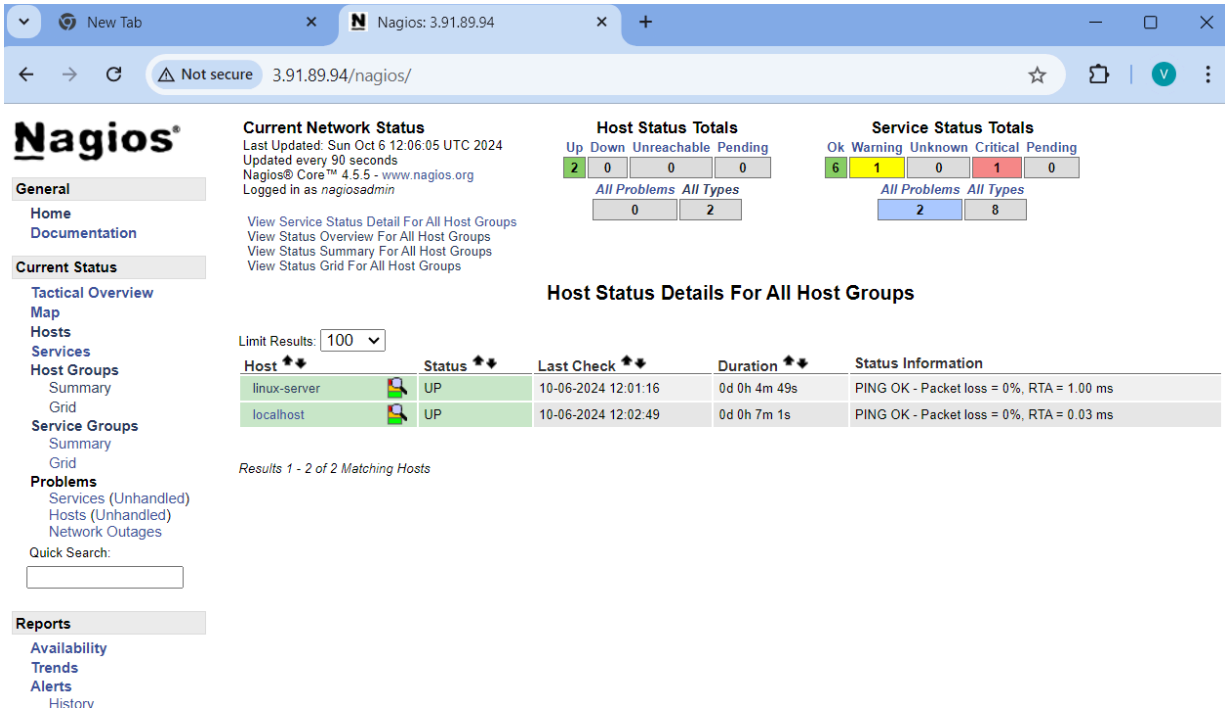
```
[root@ip-172-31-46-196 ec2-user]# sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; disabled; preset: disabled)
     Active: active (running) since Sun 2024-10-06 11:37:16 UTC; 9min ago
       Docs: https://www.nagios.org/documentation
    Process: 4481 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, st>
    Process: 4482 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, statu>
   Main PID: 4488 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 4.1M
        CPU: 108ms
     CGroup: /system.slice/nagios.service
             ├─4488 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─4489 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─4490 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─4491 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─4492 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─4497 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 06 11:37:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;1;(No output on s>
Oct 06 11:38:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;2;(No output on s>
Oct 06 11:39:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;3;(No output on s>
Oct 06 11:40:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;4;(No output on s>
Oct 06 11:41:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;5;(No output on s>
Oct 06 11:42:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;6;(No output on s>
Oct 06 11:43:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;7;(No output on s>
Oct 06 11:44:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;8;(No output on s>
Oct 06 11:45:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;SOFT;9;(No output on s>
Oct 06 11:46:16 ip-172-31-46-196.ec2.internal nagios[4488]: HOST ALERT: linux-server;DOWN;HARD;10;(No output on >
lines 1-28/28 (END)
```

sudo systemctl status httpd
sudo systemctl start httpd
sudo systemctl enable httpd

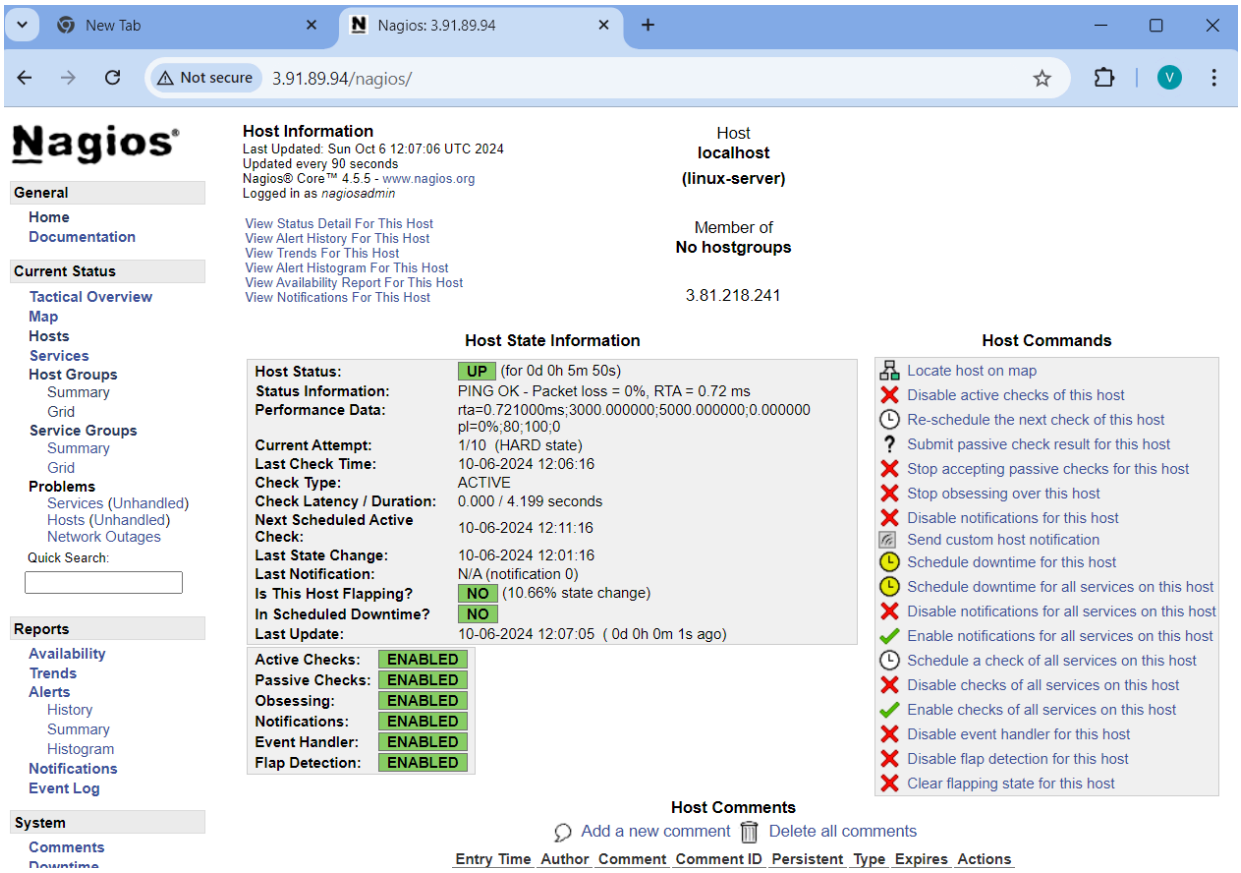
```
[root@ip-172-31-46-196 ec2-user]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/httpd.service.d
             └─php-fpm.conf
     Active: active (running) since Sun 2024-10-06 11:08:08 UTC; 42min ago
       Docs: man:httpd.service(8)
   Main PID: 2546 (httpd)
     Status: "Total requests: 48; Idle/Busy workers 100/0;Requests/sec: 0.0188; Bytes served/sec: 121 B/sec"
      Tasks: 230 (limit: 1112)
     Memory: 25.1M
        CPU: 1.834s
     CGroup: /system.slice/httpd.service
             ├─2546 /usr/sbin/httpd -DFOREGROUND
             ├─2548 /usr/sbin/httpd -DFOREGROUND
             ├─2554 /usr/sbin/httpd -DFOREGROUND
             ├─2555 /usr/sbin/httpd -DFOREGROUND
             ├─2556 /usr/sbin/httpd -DFOREGROUND
             └─2889 /usr/sbin/httpd -DFOREGROUND

Oct 06 11:08:07 ip-172-31-46-196.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Oct 06 11:08:08 ip-172-31-46-196.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Oct 06 11:08:08 ip-172-31-46-196.ec2.internal httpd[2546]: Server configured, listening on: port 80
[root@ip-172-31-46-196 ec2-user]# sudo systemctl start httpd
[root@ip-172-31-46-196 ec2-user]# sudo systemctl enable httpd
[root@ip-172-31-46-196 ec2-user]#
```

Step 15: Now to check Nagios dashboard go to http://<Nagios-host ip>/nagios .
Now Click on Hosts from left side panel

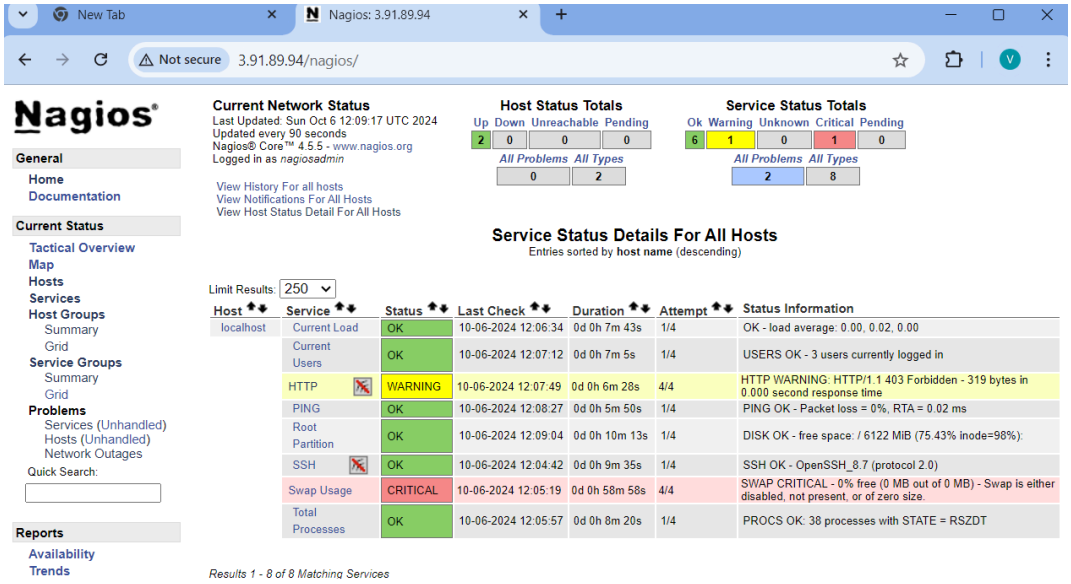
Name:Shejal Tiwari Class/Roll No: D15C/ 57



We can see our linuxserver now click on it we can see the host information.



Current Network Status



Conclusion:

In this experiment, we successfully implemented port, service, and Windows/Linux server monitoring using Nagios, but encountered a few challenges.

- **Configuration Issues:** Setting up monitoring hosts and editing files like linuxserver.cfg led to some errors in file paths and syntax, which required careful review.

- **NRPE Setup:** Configuring NRPE for remote monitoring was tricky due to firewall and permission issues, often causing connectivity problems between the Nagios host and clients.

- **Service Restarts:** Restarting Nagios and NRPE to apply changes didn't always work smoothly, with misconfigurations requiring troubleshooting.

- **Dashboard Access:** Accessing the Nagios dashboard was hindered by incorrect AWS security group rules, needing adjustments to allow proper HTTP and TCP traffic.