EXPERIMENT – 10

AIM: To perform Port, service monitoring, windows linux servers monitoring using Nagios.

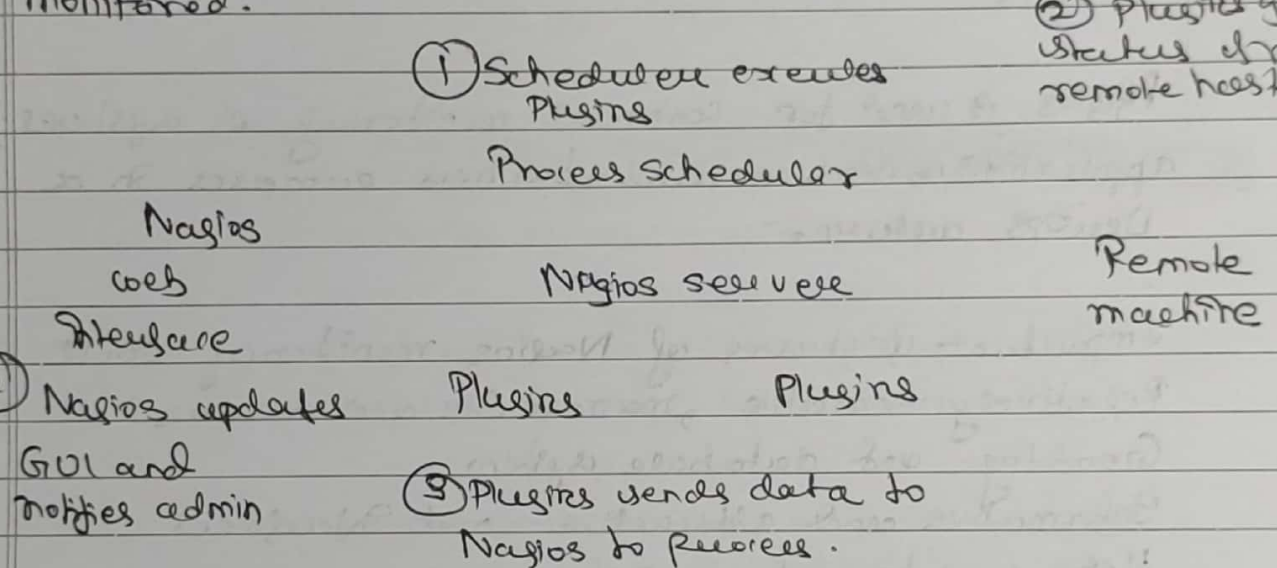
THEORY :-

Nagios is used for continous monitoring of systems, applications, service and business processes in a DevOps culture.

Important features of Nagios monitoring tool:
Relatively scalable, manageable and secure
Good log and database system
Informative and attractive web interfaces
Helps you to detect network errors or server crashes.
You can troubleshoot performance issues of server.
If any issues, can be fixed automatically as they are identified during monitoring process.
Utilizes topology to determine dependencies.
Monitor network services like HTTP, SMTP, POP etc.
Helps you to define network host hierarchy using parent hosts.
Ability to define event handlers that runs during service or host events for proactive resolution.
Support for implementing redundant monitoring hosts.

Nagios architecture

It is a client-server architecture. Usually, network, a Nagios server is running on a host and plugins are running on all remote/remote hosts which should be monitored.



Scheduler is a component of server part of Nagios. It sends a signal to execute the plugins at remote host.
The plugin gets status from remote host.
The plugin sends data to process scheduler.
The process scheduler updates the GUI and notifications are sent to admins.

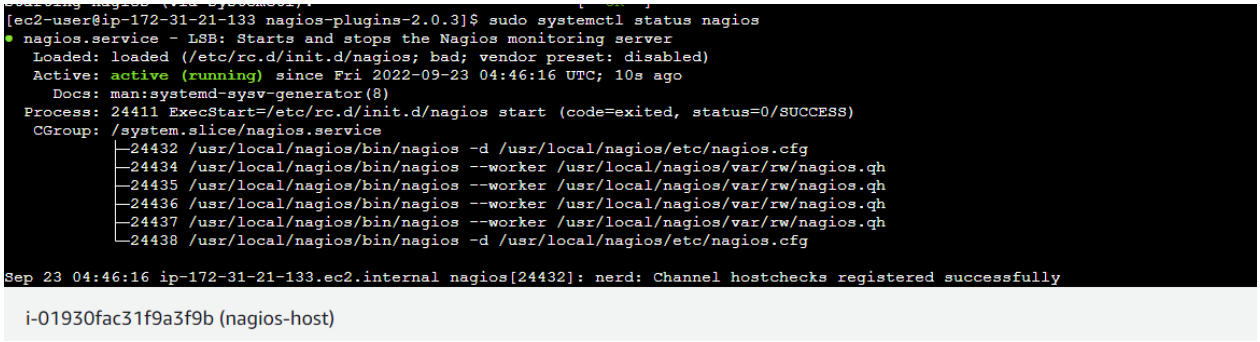
CONCLUSION:

Thus we learned about service monitoring using Nagios and successfully monitored a linux server and monitored its different ports and services using Nagios and NRPE.

Steps:

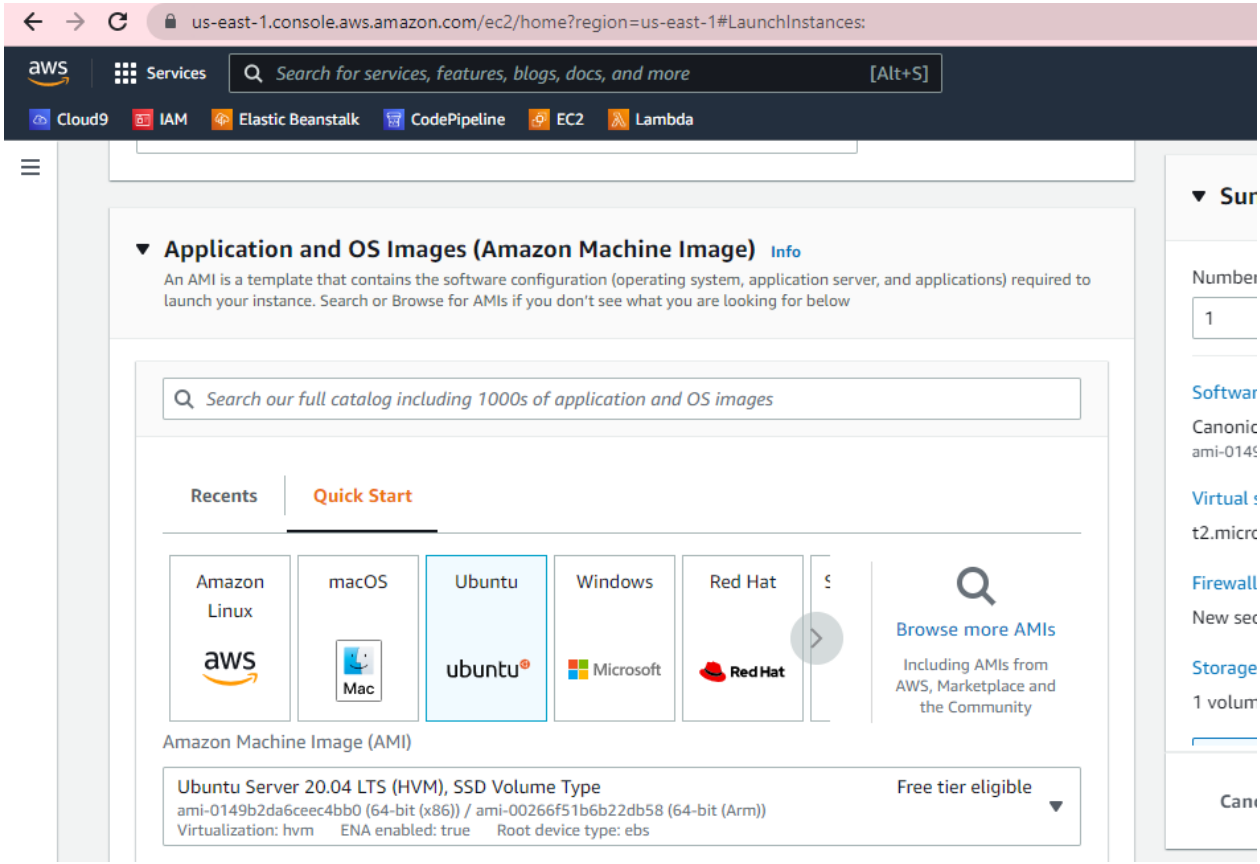
Prerequisites: AWS Free Tier, Nagios Server running on Amazon Linux Machine.

1. To Confirm that Nagios is running **on the server side**, run this *sudo systemctl status nagios* on the "NAGIOS HOST".



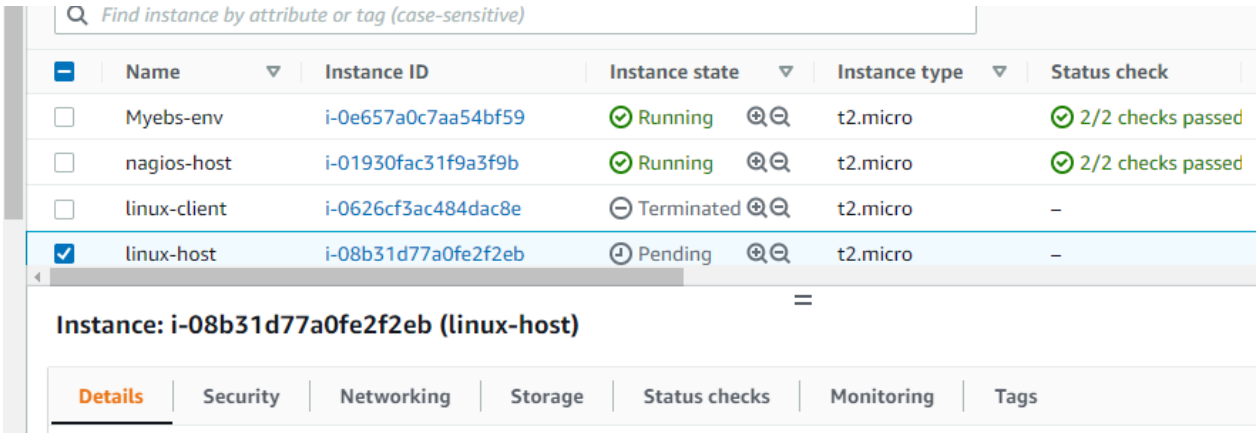
You can proceed if you get this message.

2. Before we begin,
To monitor a Linux machine, create an **Ubuntu 20.04 server EC2** Instance in AWS.



Provide it with the same security group as the Nagios Host and name it **'linux-client'** alongside the host.

For now, leave this machine as is, and go back to your nagios HOST machine.



3. On the server, run this command

```
ps -ef | grep nagios
```

```
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$ ps -ef | grep nagios
nagios    24432     1  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios    24434 24432  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios    24435 24432  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios    24436 24432  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios    24437 24432  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios    24438 24432  0 04:46 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user 24539 27314  0 04:56 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$
```

4. Become a root user and create 2 folders

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

```
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$ sudo su
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#
```

5. Copy the sample localhost.cfg file to linuxhost folder

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

```
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/li
uxserver.cfg
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#
```

6. Open linuxserver.cfg using nano and make the following changes

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

Change the hostname to linuxserver (EVERYWHERE ON THE FILE)

```
  GNU nano 2.9.8                  /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

##############################################################################
##############################################################################

# 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                 127.0.0.1
        }


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

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo  M-A Mark Text  M-] To Bracket
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo  M-6 Copy Text  M-W WhereIs Next
```

Change address to the public IP address of your **LINUX CLIENT**.

```
# 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                 107.23.206.182
        }
```

Change hostgroup_name under hostgroup to linux-servers1
Everywhere else on the file, change the hostname to linuxserver instead of localhost.

```
# 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
        }
```

7. Open the Nagios Config file and add the following line

```
nano /usr/local/nagios/etc/nagios.cfg

##Add this line
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
```

```
  GNU nano 2.9.8                                  /usr/local/nagios/etc/nagios.cfg

##############################################################################
#
# NAGIOS.CFG - Sample Main Config File for Nagios 4.0.8
#
# 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
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/


# OBJECT CONFIGURATION FILE(S)
                                [ Read 1334 lines ]
^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text  M-
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text  M-
```

8. Verify the configuration files

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

```
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-172-31-21-133 nagios-plugins-2.0.3]#
```

i-01930fac31f9a3f9b (nagios-host)

PublicIPs: 54.160.162.142 PrivateIPs: 172.31.21.133

You are good to go if there are no errors.

9. Restart the nagios service

```
service nagios restart
```

```
Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# service nagios restart
Restarting nagios (via systemctl):                         [  OK  ]
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#
```

i-01930fac31f9a3f9b (nagios-host)

Now it is time to switch to the client machine.

10. SSH into the machine or simply use the EC2 Instance Connect

    feature.

11. Make a package index update and install gcc, nagios-nrpe-server and the plugins.

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

```
Creating config file /etc/nagios-plugins/config/snmp.cfg with new version
Setting up monitoring-plugins (2.2-6ubuntu1.2) ...
Setting up python3-ldb (2:2.2.3-0ubuntu0.20.04.3) ...
Setting up libavahi-client3:amd64 (0.7-4ubuntu7.1) ...
Setting up libcups2:amd64 (2.3.1-9ubuntu1.2) ...
Setting up samba-libs:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up libsmbclient:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up smbclient (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up samba-dsdb-modules:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up python3-samba (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up samba-common-bin (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Checking smb.conf with testparm
Load smb config files from /etc/samba/smb.conf
Loaded services file OK.
Weak crypto is allowed
Server role: ROLE_STANDALONE

Done
Processing triggers for systemd (245.4-4ubuntu3.17) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
ubuntu@ip-172-31-94-97:~$
```

i-08b31d77a0fe2f2eb (linux-host)

PublicIPs: 107.23.206.182 PrivateIPs: 172.31.94.97

12. Open nrpe.cfg file to make changes.

```
sudo nano /etc/nagios/nrpe.cfg
```

Under allowed_hosts, add your nagios host IP address like so

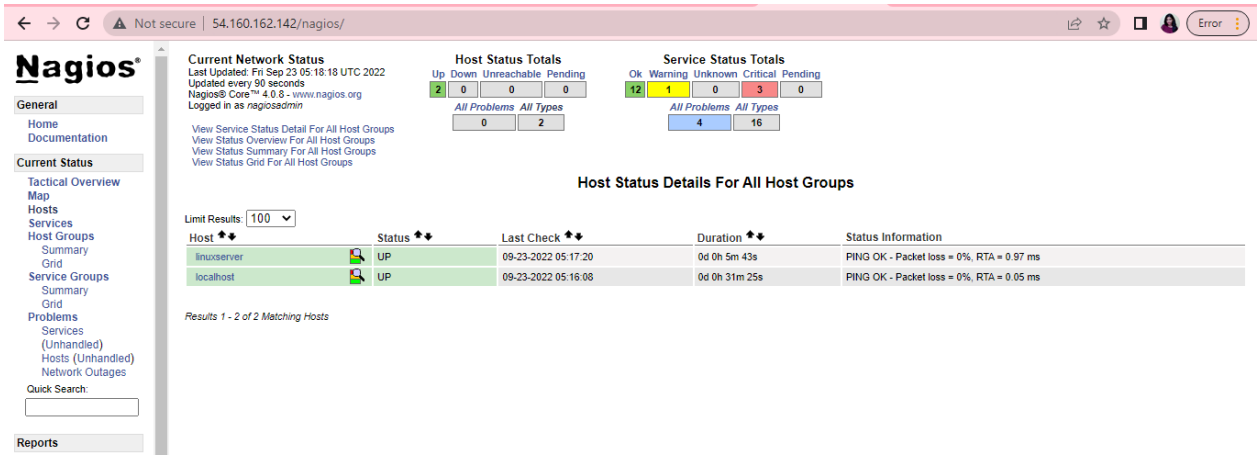
```
  GNU nano 4.8                                  /etc/nagios/nrpe.cfg
# that are allowed to talk to the NRPE daemon. Network addresses with a bit mask
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
# supported.
#
# Note: The daemon only does rudimentary checking of 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,::,54.160.162.142



# 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.
#
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```

13. Restart the NRPE server

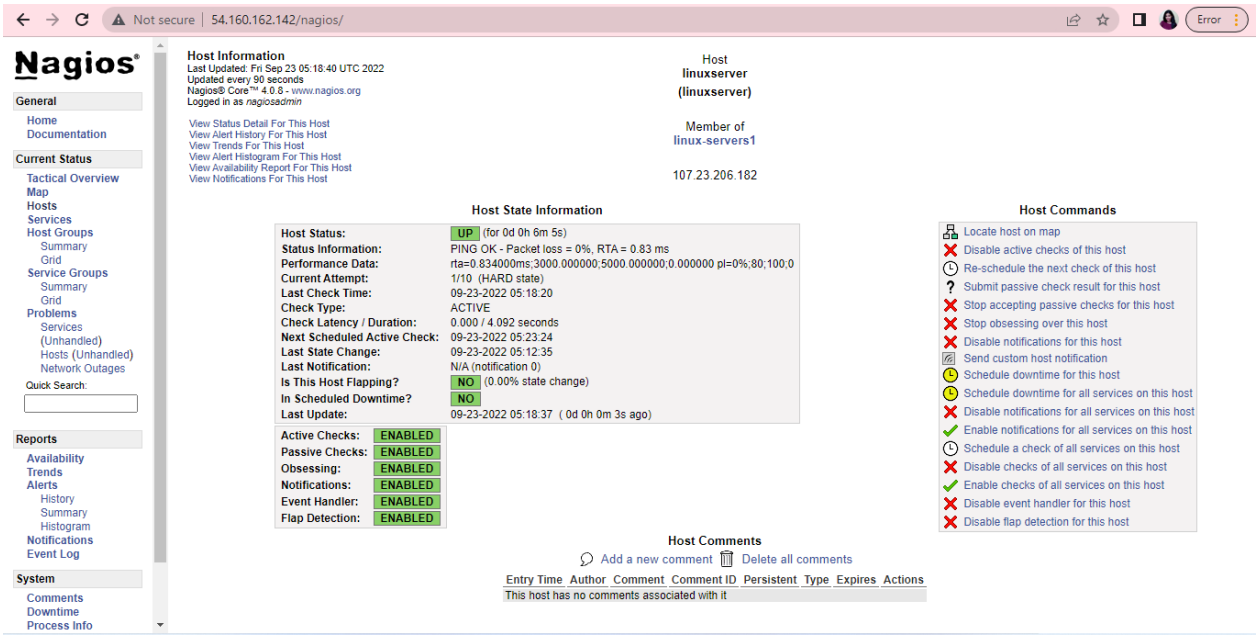
```
sudo systemctl restart nagios-nrpe-server
```

14. Now, check your nagios dashboard and you'll see a new host being added.

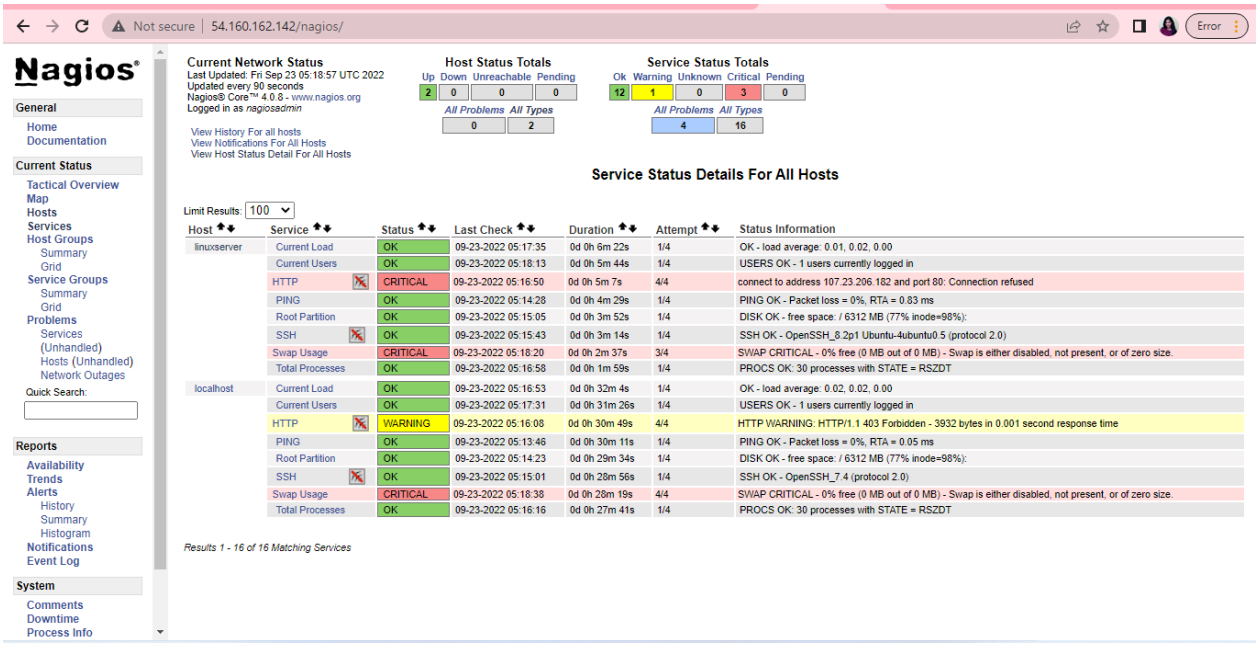
Click on Hosts.



Click on linuxserver to see the host details



You can click Services to see all services and ports being monitored.



As you can see, we have our linuxserver up and running. It is showing critical status on HTTP due to permission errors and swap because there is no partition created.

In this case, we have monitored -

Servers: 1 linux server
Services: swap
Ports: 22, 80 (ssh, http)
Processes: User status, Current load, total processes, root partition, etc.

Recommended Cleanup

- Terminate both of your EC-2 instances to avoid charges.
- Delete the security group if you created a new one (it won't affect your bill, you may avoid it)

Conclusion:

Thus, we learned about service monitoring using Nagios and successfully monitored a Linux Server and monitored its different ports and services using Nagios and NRPE.