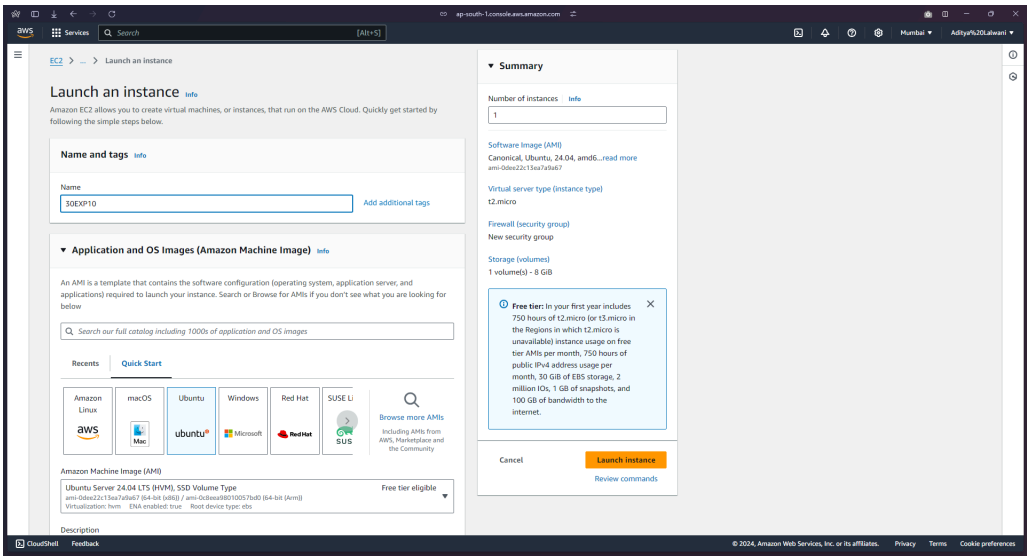
Aim: To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

Prerequisites: An Amazon Linux instance with nagios already set up.

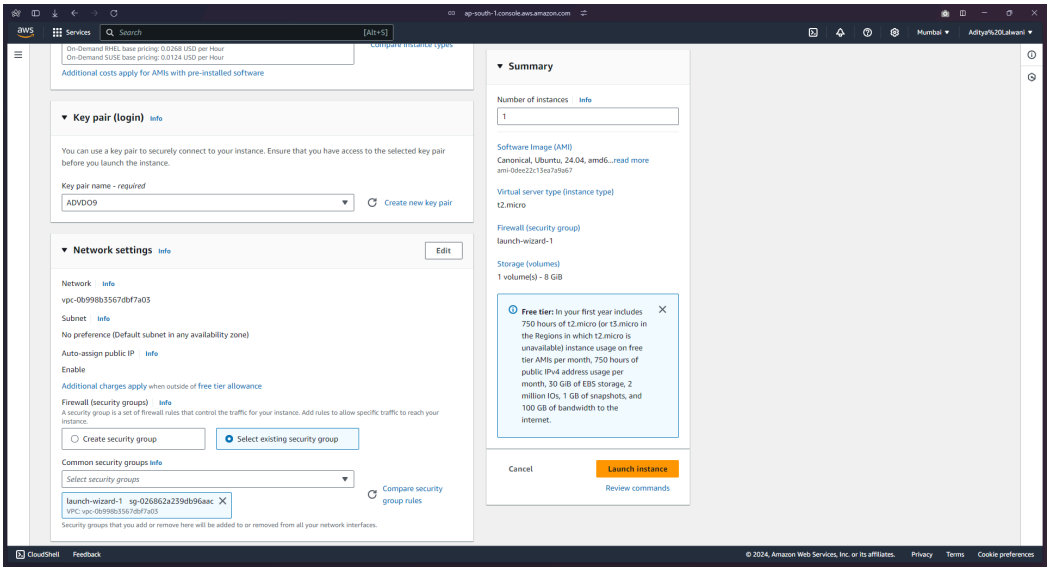
Set up ubuntu instance

Login to your AWS account. Search for EC2 on services. Open the interface and click on Create Instance.



Select The OS Image as Ubuntu.

Make sure to select the same private key that you created for the Amazon Linux instance. Also select the same security group as you created for the Linux instance.



Execute the following on Nagios Host machine (Linux)

We need to verify whether the nagios service is running or not. Fo that, run this command.

ps -ef | grep nagios

```
[ec2-user@ip-172-31-11-56 nagios-plugins-2.4.11]$ ps -ef | grep nagios
nagios     64624       1  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios     64625   64624  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     64626   64624  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     64627   64624  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     64628   64624  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     64629   64624  0 15:48 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user   65821    2337  0 16:09 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-11-56 nagios-plugins-2.4.11]$
```

Now, make yourself as the root user, and create a folder with the path /usr/local/nagios/etc/objects/monitorhosts/linuxhosts

sudo su

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

```
[ec2-user@ip-172-31-11-56 nagios-plugins-2.4.11]$ sudo su
[root@ip-172-31-11-56 nagios-plugins-2.4.11]# mkdir -p /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-11-56 nagios-plugins-2.4.11]#
```

cp

/usr/local/nagios/etc/objects/localhost.cfg

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

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

Change hostname and alias to linuxserver

Change address to public ip address of client instance (Ubuntu instance)

```
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
}
```

Change hostgroup_name to linux-servers1

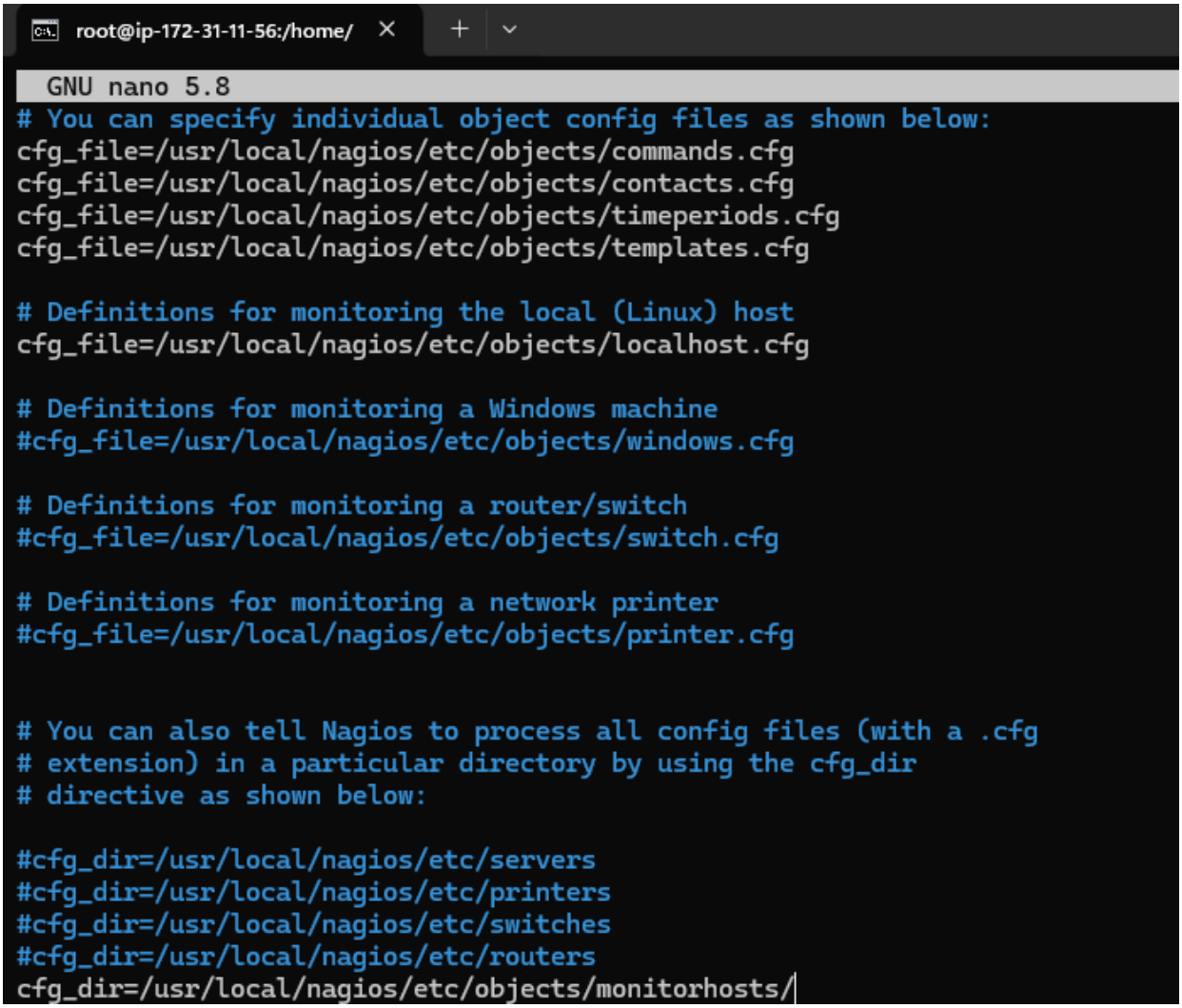
```
define hostgroup {

    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 group
}
```

Change the occurrences of hostname further in the document from localhost to linuxserver

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

add the following line

cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/



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

```
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-172-31-11-56 nagios-plugins-2.4.11]#
```

service nagios restart

```
[root@ip-172-31-83-159 nagios-plugins-2.0.3]# service nagios restart
Restarting nagios (via systemctl):                         [  OK  ]
[root@ip-172-31-83-159 nagios-plugins-2.0.3]#
```

Execute the following on Nagios Client machine (Ubuntu)

sudo apt update -y

sudo apt install gcc -y

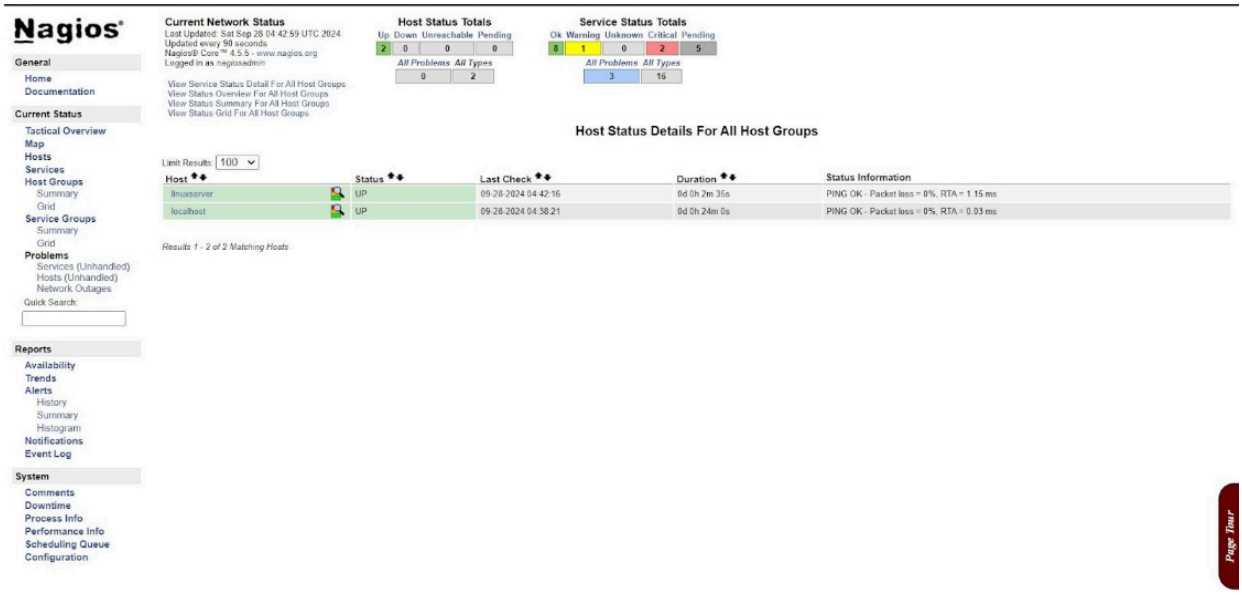
sudo apt install -y nagios-nrpe-server nagios-plugins

```
Creating config file /etc/nagios-plugins/config/snmp.cfg with new version
Setting up monitoring-plugins (2.3.5-1ubuntu3) ...
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-172-31-0-17:~$
```

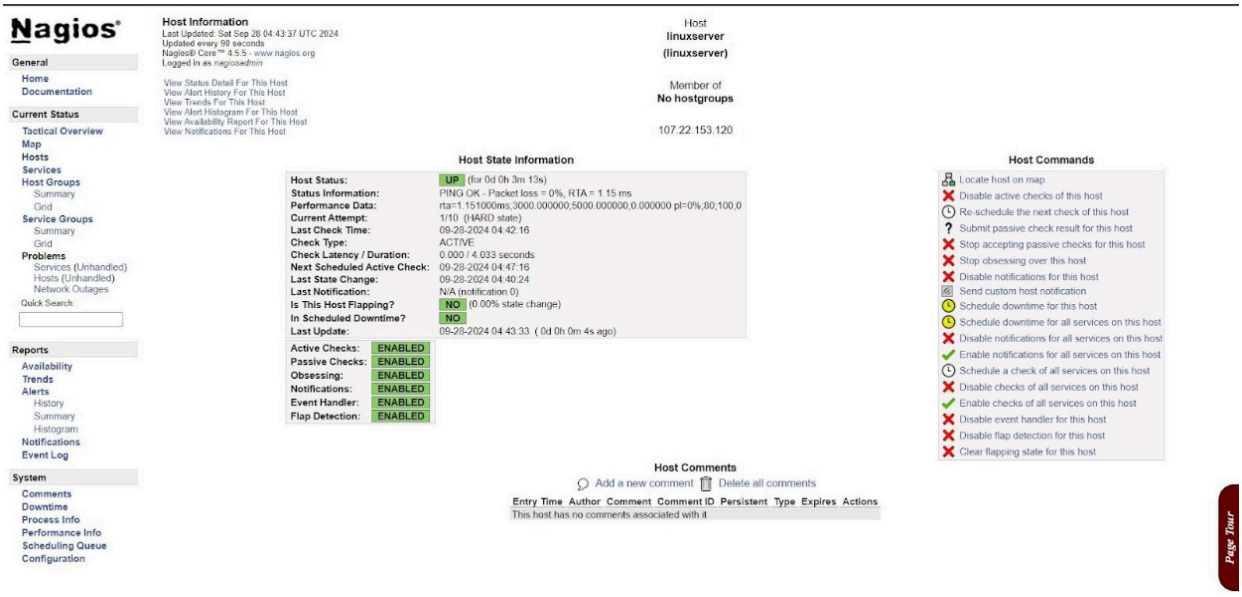
sudo nano /etc/nagios/nrpe.cfg

```
allowed_hosts=127.0.0.1,::1,35.154.231.184
```

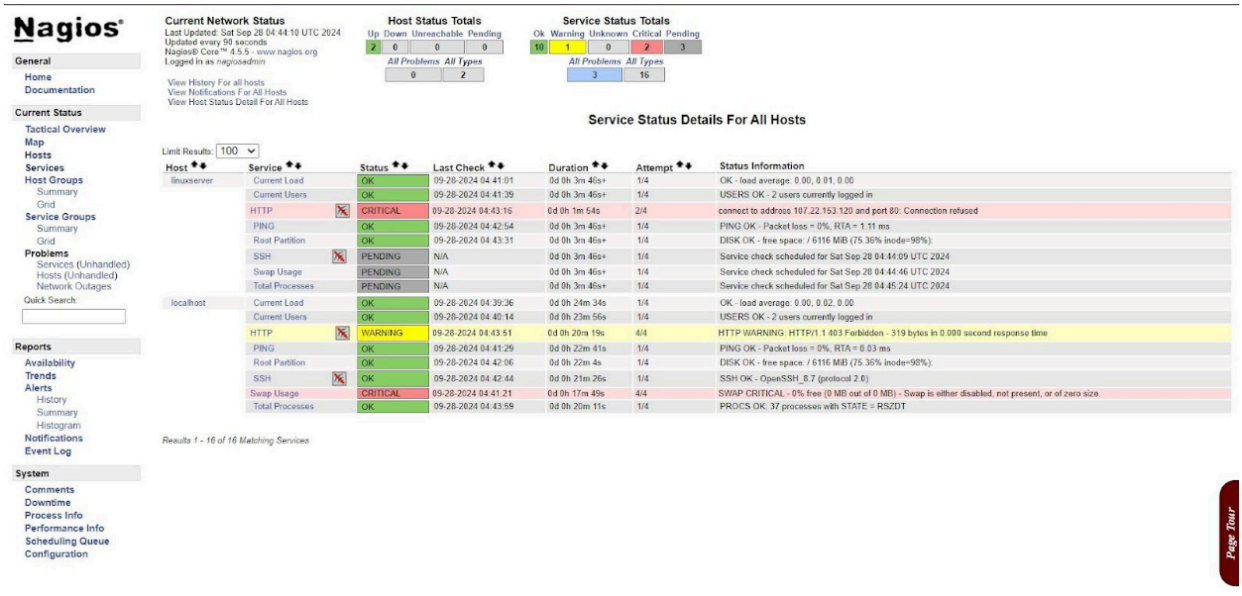
Go to Nagios dashboard, click on hosts.
the linuxserver is also added as a host.



Click on linuxserver. Here, we can check all the information about linuxserver host.



Click on services. Here we an see all the services that are being monitored by linuxserver.



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.

Conclusion:

In this experiment, we learned how to monitor port services and servers using Nagios. To do this, we used a Linux instance to host the Nagios dashboard and a separate Ubuntu instance as the second host. The process involved configuring the Linux instance and adding the IP address of the Ubuntu instance. We then replicated the initial setup on the Ubuntu instance, ensuring the IP address of the Linux instance was included in the allowed hosts. After restarting the NRPE server, we confirmed that the 'linuxserver' host had been successfully added.