EXPERIMENT NO. 10

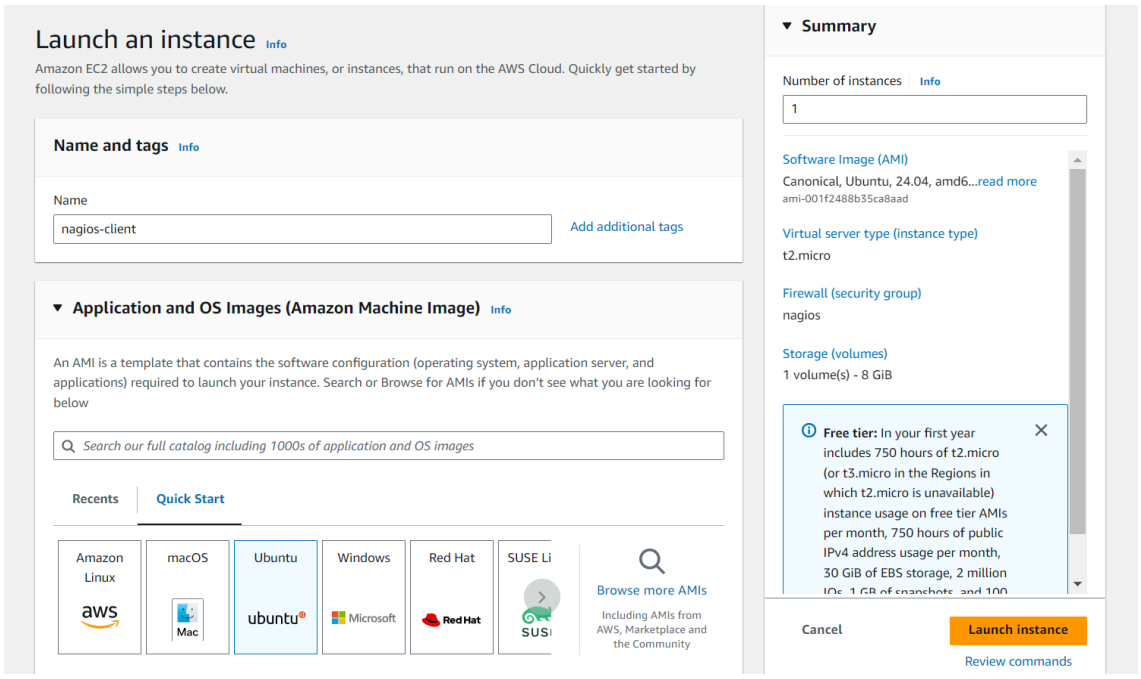
Aim: To perform Port, Service monitoring, and Windows/Linux server monitoring using Nagios.

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

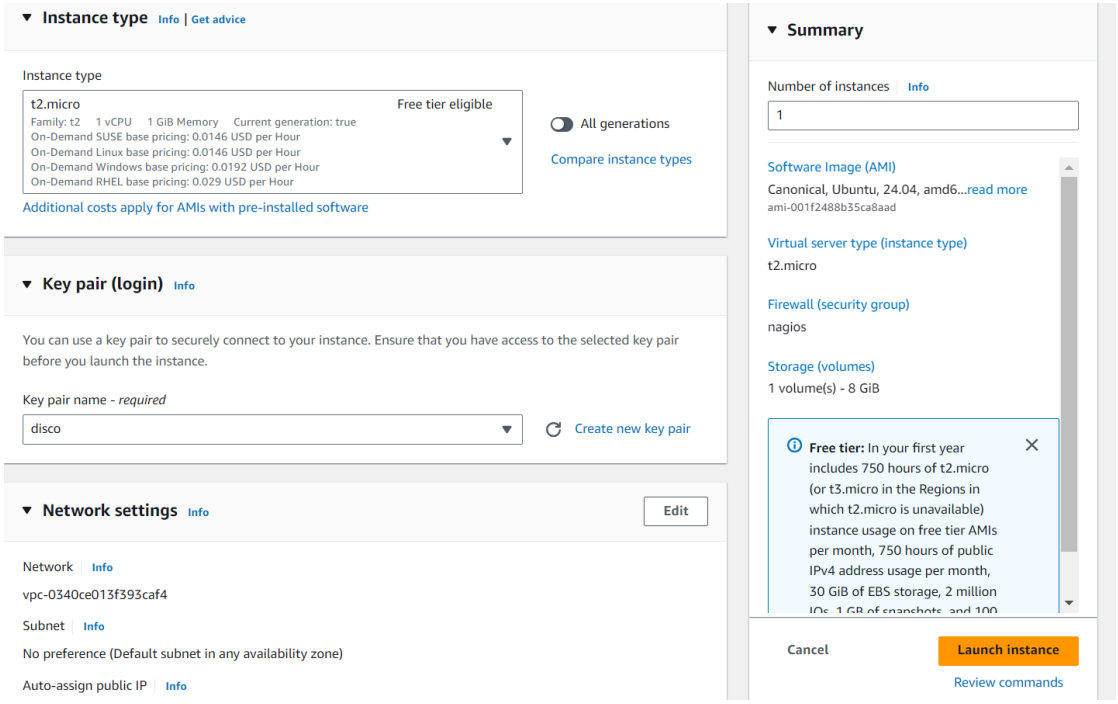
```
[ec2-user@ip-172-31-13-224 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/etc/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Sat 2024-10-05 19:22:57 UTC; 2min 1s ago
   Main PID: 74867 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 233ms
     CGroup: /system.slice/nagios.service
             ├─74867 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg
             ├─74868 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─74869 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─74870 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─74871 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─74872 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg

Oct 05 19:22:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: wproc: Registry request: name=Core Worker 74869;pid=>
Oct 05 19:22:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: wproc: Registry request: name=Core Worker 74868;pid=>
Oct 05 19:22:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: Successfully launched command file worker with pid 7>
Oct 05 19:22:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: Successfully launched command file worker with pid 7>
Oct 05 19:22:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: HOST ALERT: localhost;DOWN;SOFT;1;(No output on stdo>
Oct 05 19:23:34 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: SERVICE ALERT: localhost;Current Load;CRITICAL;HARD;>
Oct 05 19:23:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: HOST ALERT: localhost;DOWN;SOFT;2;(No output on stdo>
Oct 05 19:24:12 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: SERVICE ALERT: localhost;Current Users;CRITICAL;HARD>
Oct 05 19:24:49 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: SERVICE ALERT: localhost;HTTP;CRITICAL;HARD;1;(No ou>
Oct 05 19:24:57 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[74867]: HOST ALERT: localhost;DOWN;SOFT;3;(No output on stdo>
lines 1-25/25 (END)
```

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

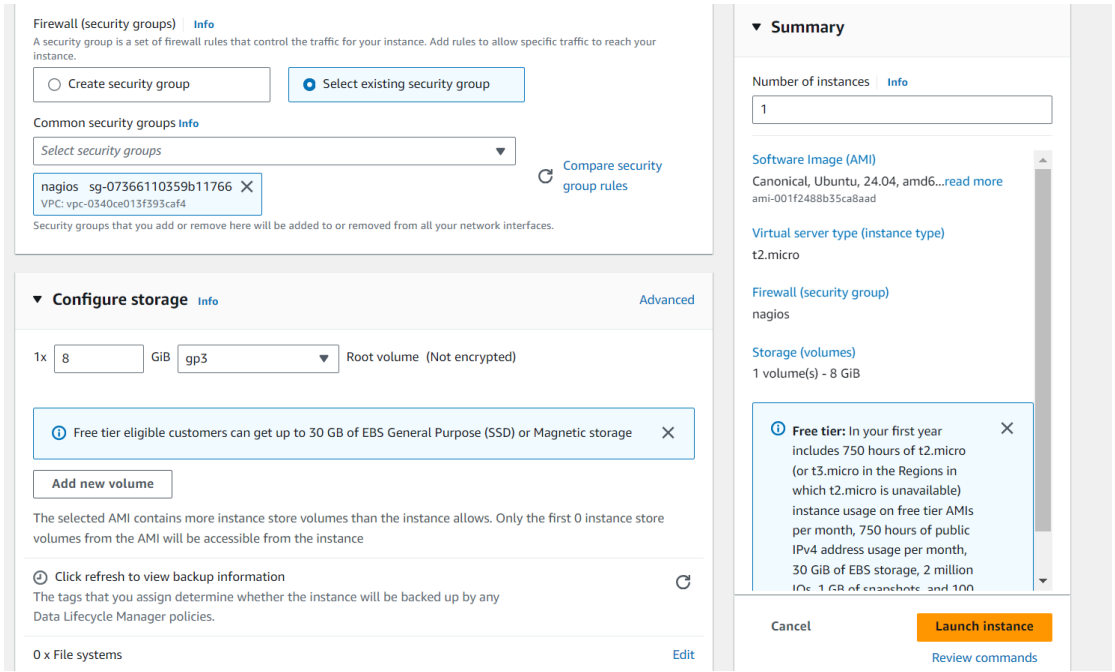
2. Now Create a new EC2 instance. Name: Nagios-client, AMI: Ubuntu Instance Type: t2.micro.



For Key pair : Click on create key and make key of type RSA with extension .pem . Key will be downloaded to your local machine. Now select that key in the key pair if you already have a key with type RSA and extension .pem no need to create a new key but you must have that key downloaded.



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).



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.

```
PS C:\Users\bhumi> cd "C:\Users\bhumi\OneDrive\Desktop\New folder"
PS C:\Users\bhumi\OneDrive\Desktop\New folder> ssh -i "disco.pem" ubuntu@ec2-3-25-84-91.ap-southeast-2.compute.amazonaws.com
The authenticity of host 'ec2-3-25-84-91.ap-southeast-2.compute.amazonaws.com (3.25.84.91)' can't be established.
ED25519 key fingerprint is SHA256:zIXx3ATmWCr9U4e6inijfig4vG+Bji+Xm8lz0cDdHCc.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-25-84-91.ap-southeast-2.compute.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 Sat Oct  5 18:51:05 UTC 2024

  System load:  0.0               Processes:             105
  Usage of /:   23.0% of 6.71GB   Users logged in:       0
  Memory usage: 20%               IPv4 address for enX0: 172.31.5.17
  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

Last login: Sat Oct  5 18:26:23 2024 from 13.239.158.3
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

Successfully connected to the instance.

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

4. Now on the server Nagios-host run the following command.
   ps -ef | grep nagios

```
[ec2-user@ip-172-31-13-224 ~]$ ps -ef | grep nagios
nagios     74867       1  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg
nagios     74868   74867  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     74869   74867  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     74870   74867  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     74871   74867  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios     74872   74867  0 19:22 ?        00:00:00 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg
ec2-user   75110   75013  0 19:26 pts/1    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-13-224 ~]$
```

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-172-31-13-224 ~]$ sudo su
[root@ip-172-31-13-224 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-13-224 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-13-224 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-172-31-13-224 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhost
s/linuxserver.cfg
[root@ip-172-31-13-224 ec2-user]#
```

7. Open linuxserver.cfg using nano and make the following changes everywhere in the 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-172-31-13-224 ec2-user]# sudo sed -i 's/^ *host_name.*/    host_name       linuxserver/' /usr/local/nagios/etc/objects/moni
torhosts/linuxhosts/linuxserver.cfg
sudo sed -i 's/^ *address.*/    address         3.25.70.155/' /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
sudo sed -i 's/^ *hostgroup_name.*/    hostgroup_name   linux-servers1/' /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxs
erver.cfg
[root@ip-172-31-13-224 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-13-224 ec2-user]# nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

```
##############################################################################
#
# 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 that are defined
                                                    ; in (or inherited by) the linux-server host template definition.
    host_name       linuxserver
    alias                   localhost
    address         3.25.70.155
}
```

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

```
# 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/
```

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-172-31-13-224 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...
   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
```

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

```
[root@ip-172-31-13-224 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-13-224 ec2-user]#
```

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-172-31-5-17:~$ sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins
Hit:1 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [537 kB]
Get:14 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [132 kB]
Get:15 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [8860 B]
Get:16 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [384 kB]
Get:17 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [159 kB]
```

```
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-172-31-5-17:~$
```

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

```
# 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,::1,3.25.70.155
```

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

```
ubuntu@ip-172-31-5-17:~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-5-17:~$
```

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 activate it.
    sudo systemctl status nagios

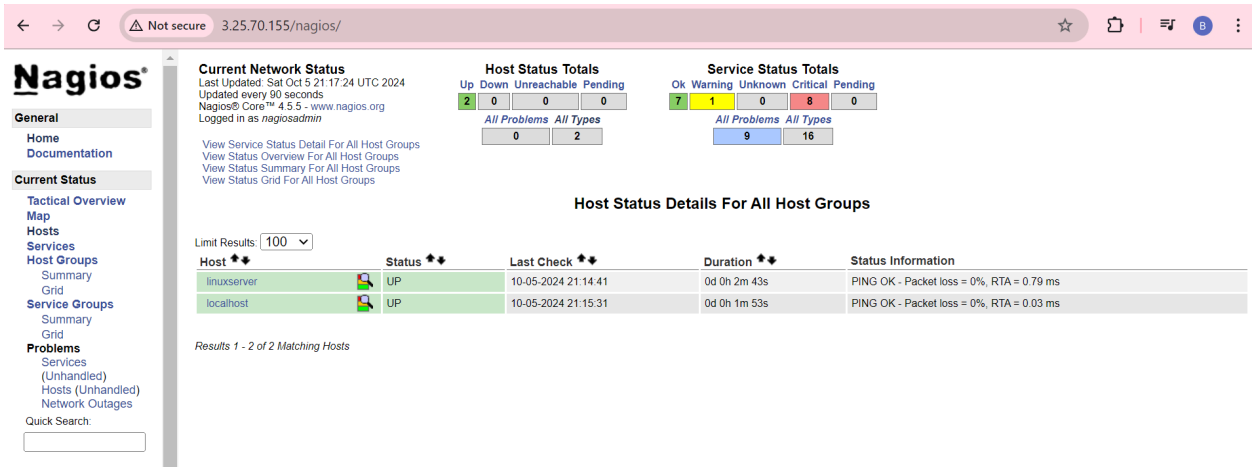
```
[root@ip-172-31-13-224 ec2-user]# sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
     Loaded: loaded (/etc/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Sat 2024-10-05 19:42:04 UTC; 22min ago
   Main PID: 76185 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 4.1M
        CPU: 252ms
     CGroup: /system.slice/nagios.service
             ├─76185 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg
             ├─76186 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─76187 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─76188 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─76189 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─76190 /usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg

Oct 05 19:45:11 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: SERVICE ALERT: linuxserver;Root Partition;CRITICAL;HARD;1;(No o>
Oct 05 19:45:49 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: SERVICE ALERT: linuxserver;SSH;CRITICAL;HARD;1;(No output on st>
Oct 05 19:46:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;SOFT;5;(No output on stdout) stder>
Oct 05 19:46:26 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: SERVICE ALERT: linuxserver;Swap Usage;CRITICAL;HARD;1;(No outpu>
Oct 05 19:47:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: SERVICE ALERT: linuxserver;Total Processes;CRITICAL;HARD;1;(No >
Oct 05 19:47:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;SOFT;6;(No output on stdout) stder>
Oct 05 19:48:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;SOFT;7;(No output on stdout) stder>
Oct 05 19:49:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;SOFT;8;(No output on stdout) stder>
Oct 05 19:50:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;SOFT;9;(No output on stdout) stder>
Oct 05 19:51:04 ip-172-31-13-224.ap-southeast-2.compute.internal nagios[76185]: HOST ALERT: linuxserver;DOWN;HARD;10;(No output on stdout) stde>
lines 1-25/25 (END)
```

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

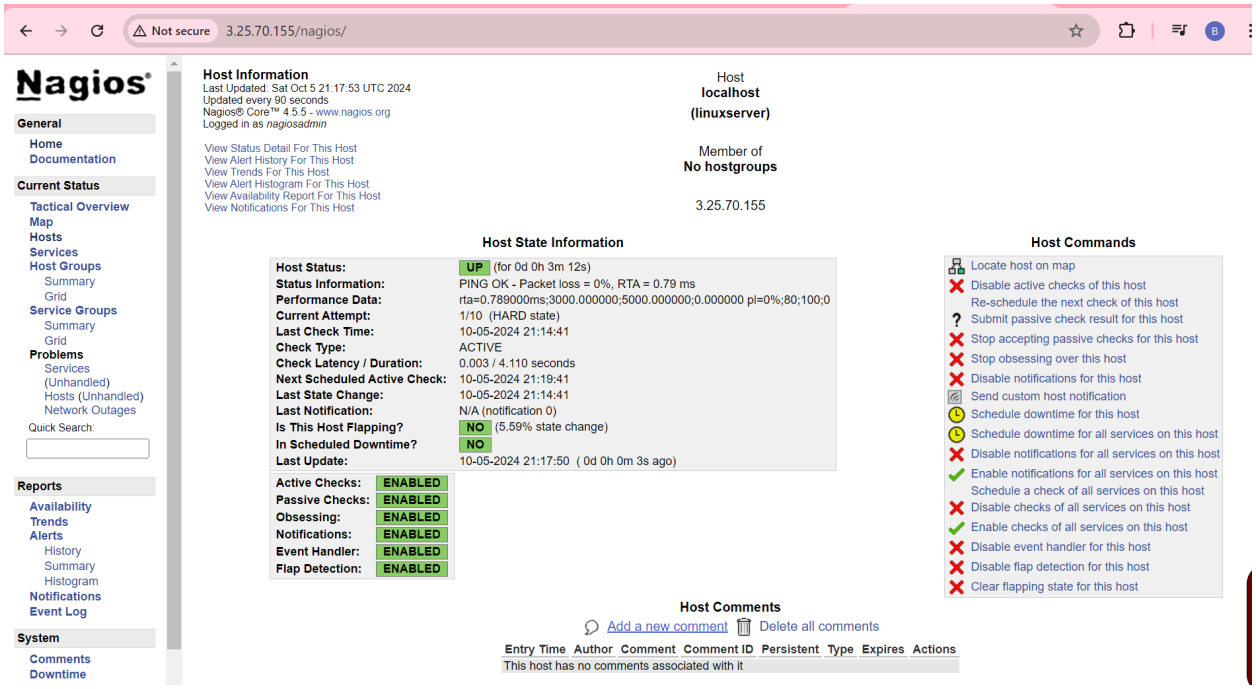
```
[root@ip-172-31-13-224 ec2-user]# sudo systemctl status httpd
sudo systemctl start httpd
sudo systemctl enable httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/httpd.service.d
             └─php-fpm.conf
     Active: active (running) since Sat 2024-10-05 19:12:34 UTC; 51min ago
       Docs: man:httpd.service(8)
   Main PID: 49647 (httpd)
     Status: "Total requests: 24; Idle/Busy workers 100/0;Requests/sec: 0.00772; Bytes served/sec:  52 B/sec"
      Tasks: 230 (limit: 1112)
     Memory: 21.5M
        CPU: 1.806s
     CGroup: /system.slice/httpd.service
             ├─49647 /usr/sbin/httpd -DFOREGROUND
             ├─49654 /usr/sbin/httpd -DFOREGROUND
             ├─49655 /usr/sbin/httpd -DFOREGROUND
             ├─49656 /usr/sbin/httpd -DFOREGROUND
             ├─49657 /usr/sbin/httpd -DFOREGROUND
             └─77500 /usr/sbin/httpd -DFOREGROUND

Oct 05 19:12:34 ip-172-31-13-224.ap-southeast-2.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Oct 05 19:12:34 ip-172-31-13-224.ap-southeast-2.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Oct 05 19:12:34 ip-172-31-13-224.ap-southeast-2.compute.internal httpd[49647]: Server configured, listening on: port 80
[root@ip-172-31-13-224 ec2-user]#
```

15. Now to check Nagios dashboard go to http://<nagios-host-public-ip>/nagios.

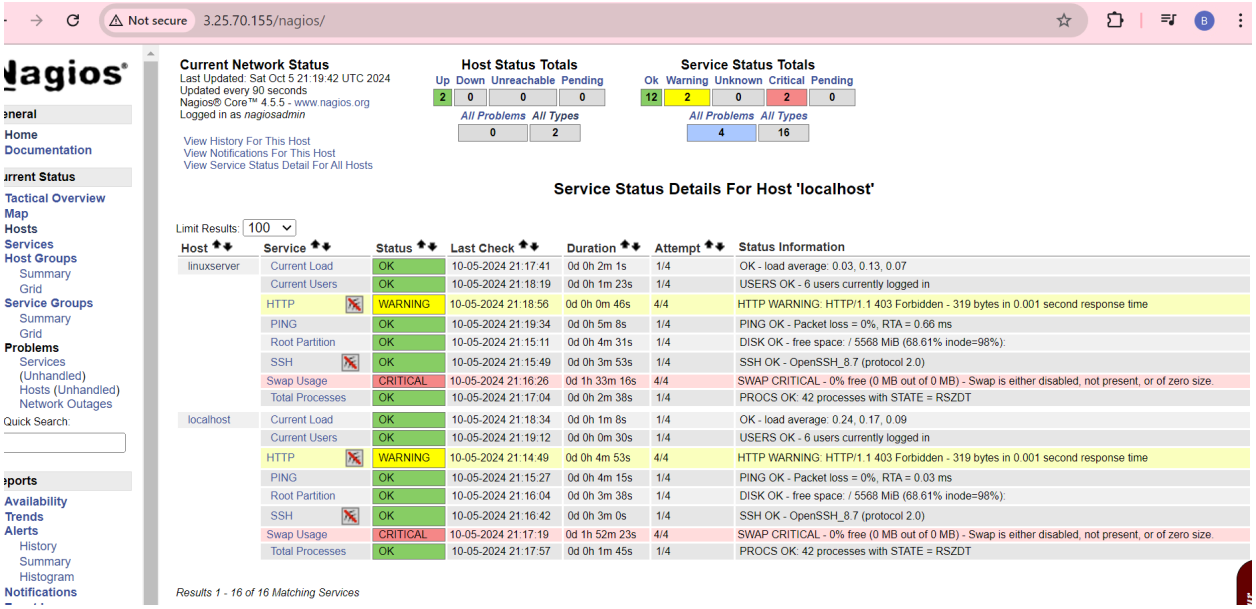
Now Click on Hosts from left side panel



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



Current Network Status



Conclusion: In conclusion, the experiment focused on monitoring ports, services, and a Linux server using Nagios. Through the step-by-step process, we successfully configured Nagios to monitor essential network services on the Linux server. By setting up both the Nagios host and client, we were able to track system performance, ensure service availability, and monitor key metrics like CPU and memory usage.