

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).

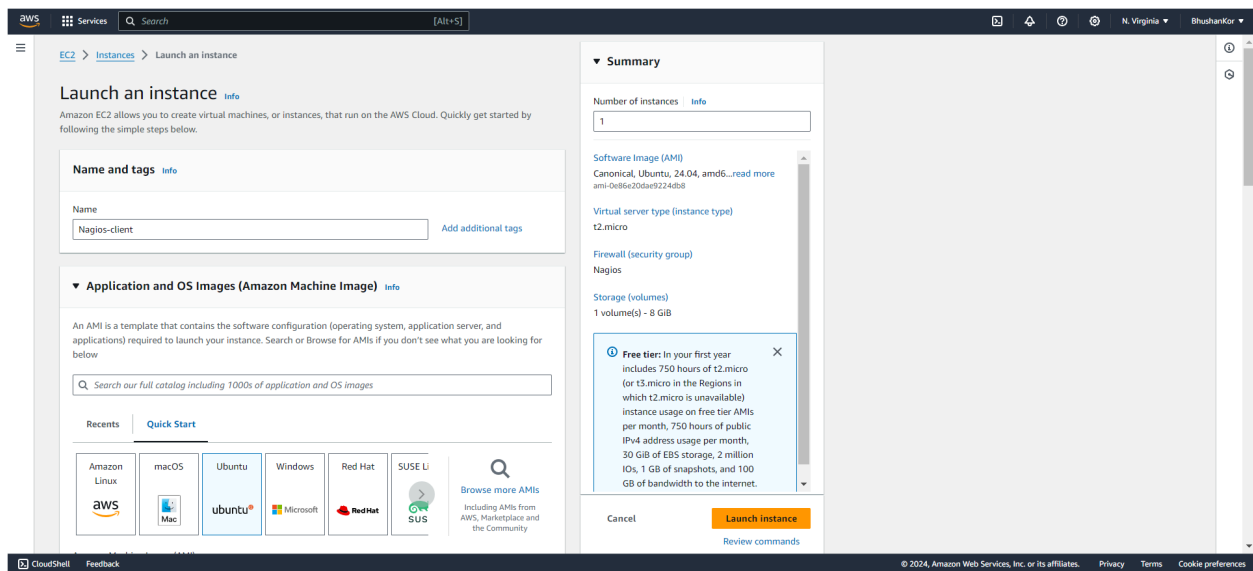
sudo systemctl status nagios

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.

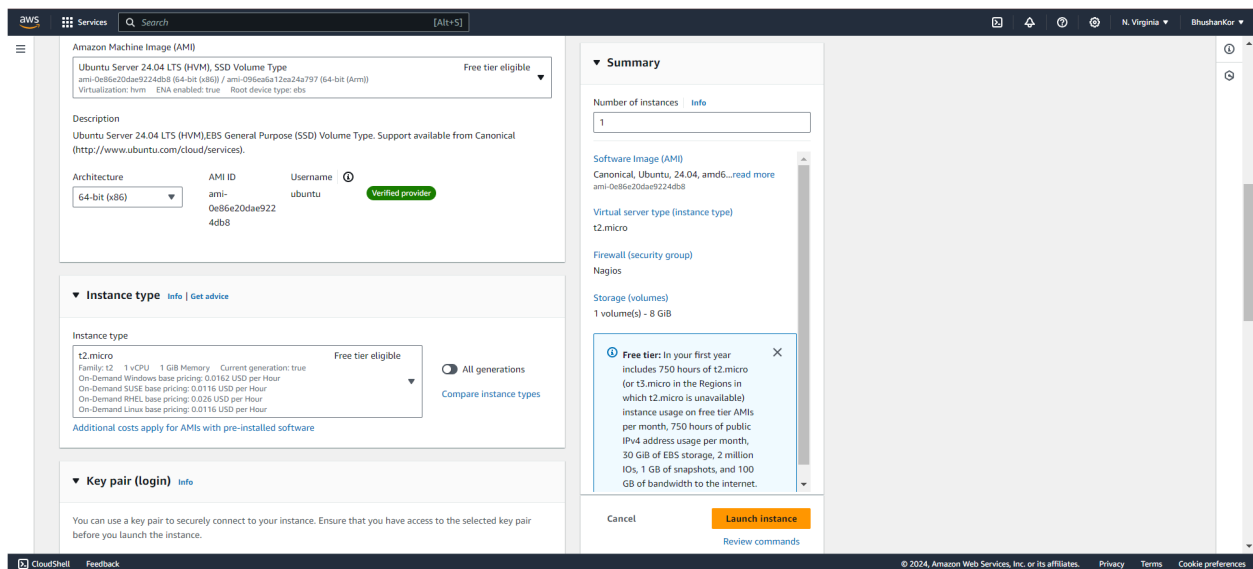
```
● nagios.service - Nagios Core 4.5.5
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Fri 2024-10-04 14:11:55 UTC; 28min ago
     Docs: https://www.nagios.org/documentation
    Main PID: 1998 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 6.7M
        CPU: 442ms
    CGroup: /system.slice/nagios.service
            └─1998 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              └─2004 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                └─2005 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  └─2006 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                    └─2007 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                      └─2008 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 14:11:55 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: Registry request: name=Core Worker>
Oct 04 14:11:55 ip-172-31-34-108.ec2.internal nagios[1998]: Successfully launched command file worker>
Oct 04 14:13:47 ip-172-31-34-108.ec2.internal nagios[1998]: SERVICE ALERT: localhost;HTTP;CRITICAL;HA>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: SERVICE NOTIFICATION: nagiosadmin;localho>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: NOTIFY job 2 from worker Core Work>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: host=localhost; service=Swap Usa>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: early_timeout=0; exited_ok=1; wa>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: stderr line 01: /bin/sh: line 1:>
Oct 04 14:16:17 ip-172-31-34-108.ec2.internal nagios[1998]: wproc: stderr line 02: /usr/bin/printf:>
Oct 04 14:18:47 ip-172-31-34-108.ec2.internal nagios[1998]: SERVICE ALERT: localhost;HTTP;WARNING;HAR>
lines 1-26/26 (END)
```

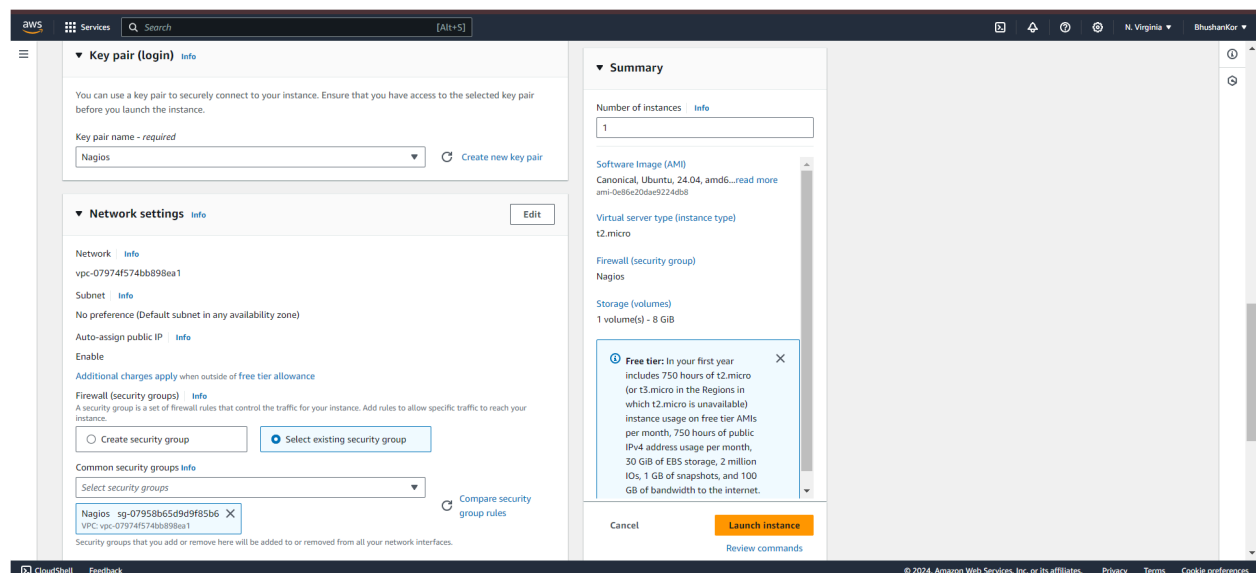


For Key pair : Click on create key and make key of type RSA with extension .pem . Key will be downloaded to your local machine.

Now select that key in key pair if you already have key with type RSA and extension .pem no need to create 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).



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\hp\Downloads\lab09> ssh -i "lab09.pem" ec2-user@ec2-54-172-217-167.compute-1.amazonaws.com
The authenticity of host 'ec2-54-172-217-167.compute-1.amazonaws.com (54.172.217.167)' can't be established.
ED25519 key fingerprint is SHA256:LBNecYQvwiUAFbRZ4UavnvY9vHuBhVKS/K3DVKJ9Jn4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-172-217-167.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

#_
~\_##### Amazon Linux 2023
~~~\#####
~~~\###|
~~~\#/ --- https://aws.amazon.com/linux/amazon-linux-2023
~~~V~' ' ->
~~~~
~~~.-'
~~~_/_/_/_/_
~~~_/_/_/_/_
[ec2-user@ip-172-31-38-4 ~]$ |
```

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-172-31-34-108 ~]$ ps -ef | grep nagios
nagios      1998      1  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios      2004     1998  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2005     1998  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2006     1998  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2007     1998  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      2008     1998  0 14:11 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
root        16967    2370  0 14:20 pts/0    00:00:00 sudo systemctl status nagios
root        16970    16967  0 14:20 pts/1    00:00:00 sudo systemctl status nagios
root        16971    16970  0 14:20 pts/1    00:00:00 systemctl status nagios
root        18221    18132  0 14:40 pts/2    00:00:00 sudo systemctl status nagios
root        18223    18221  0 14:40 pts/3    00:00:00 sudo systemctl status nagios
root        18224    18223  0 14:40 pts/3    00:00:00 systemctl status nagios
ec2-user    19275    19251  0 14:59 pts/4    00:00:00 grep --color=auto nagios
```

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-172-31-34-108 ~]$ sudo su
mkdir /usr/local/nagios/etc/objects/monitorhosts
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-34-108 ec2-user]# |
```

```
[root@ip-172-31-34-108 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-34-108 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

Step 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-172-31-34-108 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-34-108 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

```
define host {  
  
    use                linux-server          ; Name of host template to use  
                                           ; This host definition will inheri>  
                                           ; in (or inherited by) the linux-s>  
  
    host_name          linuxserver  
    alias              localhost  
    address            54.172.217.167  
}
```

```
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 th>  
}
```

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`

```
# 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
```

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-172-31-34-108 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/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 152)
Warning: Duplicate definition found for service 'SSH' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 138)
Warning: Duplicate definition found for service 'Swap Usage' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 125)
Warning: Duplicate definition found for service 'Current Load' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 112)
Warning: Duplicate definition found for service 'Total Processes' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 100)
Warning: Duplicate definition found for service 'Current Users' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 86)
Warning: Duplicate definition found for service 'Root Partition' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', starting on line 72)
Warning: Duplicate definition found for service 'PING' on host 'localhost' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg', 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
```

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

```
Things look okay - no serious problems were detected during
[root@ip-172-31-34-108 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-34-108 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-172-31-47-124:~$ 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://security.ubuntu.com/ubuntu noble-security/main Translation-en [83.9 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4704 B]
Get:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [277 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [117 kB]
Get:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]
Get:13 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [10.4 kB]
Get:14 http://security.ubuntu.com/ubuntu noble-security/multiverse
```

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

sudo nano /etc/nagios/nrpe.cfg

```
GNU nano 7.2 /etc/nagios/nrpe.cfg
#
# NOTE: This option is ignored if NRPE is running under either in>
nrpe_user=nagios

# NRPE GROUP
# This determines the effective group that the NRPE daemon should >
# You can either supply a group name or a GID.
#
# NOTE: This option is ignored if NRPE is running under either in>
nrpe_group=nagios

# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostna>
# that are allowed to talk to the NRPE daemon. Network addresses w>
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are>
# supported.
#
# Note: The daemon only does rudimentary checking of the client's >
# address. I would highly recommend adding entries in your /etc/h>
# 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 in>
allowed_hosts=127.0.0.1,::1,54.161.62.217
```

Step 13: Now restart the NRPE server by this command.

sudo systemctl restart nagios-nrpe-server

```
ubuntu@ip-172-31-47-124:~$ sudo systemctl restart nagios-nrpe-serve
r
```

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


```
[ec2-user@ip-172-31-34-108 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Fri 2024-10-04 15:39:38 UTC; 14min ago
     Docs: https://www.nagios.org/documentation
   Process: 21775 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg >
   Process: 21776 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (co
 Main PID: 21781 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 4.1M
      CPU: 244ms
   CGroup: /system.slice/nagios.service
           └─21781 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─21782 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─21783 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                 └─21784 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─21785 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                     └─21790 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Warning: Duplicate definition found >
Oct 04 15:39:38 ip-172-31-34-108.ec2.internal nagios[21781]: Successfully launched command file w>
Oct 04 15:40:20 ip-172-31-34-108.ec2.internal nagios[21781]: HOST ALERT: linuxserver;UP;SOFT;1;PI>
lines 1-28/28 (END)
```

sudo systemctl status httpd

```
[ec2-user@ip-172-31-34-108 ~]$ 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 Fri 2024-10-04 14:18:18 UTC; 1h 37min ago
     Docs: man:httpd.service(8)
 Main PID: 2495 (httpd)
   Status: "Total requests: 48; Idle/Busy workers 100/0; Requests/sec: 0.0082; Bytes served/sec:>
    Tasks: 230 (limit: 1112)
   Memory: 24.0M
      CPU: 3.655s
   CGroup: /system.slice/httpd.service
           └─2495 /usr/sbin/httpd -DFOREGROUND
             └─2543 /usr/sbin/httpd -DFOREGROUND
               └─2544 /usr/sbin/httpd -DFOREGROUND
                 └─2545 /usr/sbin/httpd -DFOREGROUND
                   └─2546 /usr/sbin/httpd -DFOREGROUND
                     └─20138 /usr/sbin/httpd -DFOREGROUND

Oct 04 14:18:18 ip-172-31-34-108.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP>
Oct 04 14:18:18 ip-172-31-34-108.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP>
Oct 04 14:18:18 ip-172-31-34-108.ec2.internal httpd[2495]: Server configured, listening on: port >
lines 1-22/22 (END)
```

sudo systemctl start httpd

sudo systemctl enable httpd

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

Nagios® Core™
✓ Daemon running with PID 21781

Nagios® Core™
Version 4.5.5
September 17, 2024
[Check for updates](#)

Get Started

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

Quick Links

- [Nagios Library](#) (tutorials and docs)
- [Nagios Labs](#) (development blog)
- [Nagios Exchange](#) (plugins and addons)
- [Nagios Support](#) (tech support)
- [Nagios.com](#) (company)
- [Nagios.org](#) (project)

Latest News

Don't Miss...

Copyright © 2010-2024 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, servicemarks, registered trademarks

Now Click on Hosts from left side panel

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

Current Network Status

Current Network Status
Last Updated: Fri Oct 4 15:58:07 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - [www.nagios.org](#)
Logged in as nagiosadmin

View Service Status Detail For All Host Groups
View Status Overview For All Host Groups
View Status Summary For All Host Groups
View Status Grid For All Host Groups

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
6	1	0	1	0

Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	10-04-2024 15:55:16	0d 0h 17m 51s	PING OK - Packet loss = 0%, RTA = 1.31 ms
localhost	UP	10-04-2024 15:53:47	0d 11h 10m 6s	PING OK - Packet loss = 0%, RTA = 0.03 ms

Results 1 - 2 of 2 Matching Hosts

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

Host Information

Last Updated: Fri Oct 4 15:58:45 UTC 2024
Updated every 90 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin

[View Status Detail For This Host](#)
[View Alert History For This Host](#)
[View Trends For This Host](#)
[View Alert Histogram For This Host](#)
[View Availability Report For This Host](#)
[View Notifications For This Host](#)

Host
localhost
(linuxserver)

Member of
No hostgroups

34.207.171.220

Host State Information

Host Status: **UP** (for 0d 0h 18m 29s)
Status Information: PING OK - Packet loss = 0%, RTA = 1.31 ms
Performance Data: rta=1.314000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0
Current Attempt: 1/10 (HARD state)
Last Check Time: 10-04-2024 15:55:16
Check Type: ACTIVE
Check Latency / Duration: 0.002 / 4.012 seconds
Next Scheduled Active Check: 10-04-2024 16:00:16
Last State Change: 10-04-2024 15:40:16
Last Notification: N/A (notification 0)
Is This Host Flapping? **NO** (10.79% state change)
In Scheduled Downtime? **NO**
Last Update: 10-04-2024 15:58:37 (0d 0h 0m 8s ago)
Active Checks: **ENABLED**
Passive Checks: **ENABLED**
Obsessing: **ENABLED**
Notifications: **ENABLED**
Event Handler: **ENABLED**
Flap Detection: **ENABLED**

Host Commands

- Locate host on map
- Disable active checks of this host
- Re-schedule the next check of this host
- Submit passive check result for this host
- Stop accepting passive checks for this host
- Stop obsessing over this host
- Disable notifications for this host
- Send custom host notification
- Schedule downtime for this host
- Schedule downtime for all services on this host
- Disable notifications for all services on this host
- Enable notifications for all services on this host
- Schedule a check of all services on this host
- Disable checks of all services on this host
- Enable checks of all services on this host
- Disable event handler for this host
- Disable flap detection for this host
- Clear flapping state for this host

Host Comments

Add a new comment Delete all comments

Entry Time	Author	Comment	Comment ID	Persistent	Type	Expires	Actions
This host has no comments associated with it							

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.