

Adv DevOps Exp 10

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

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).

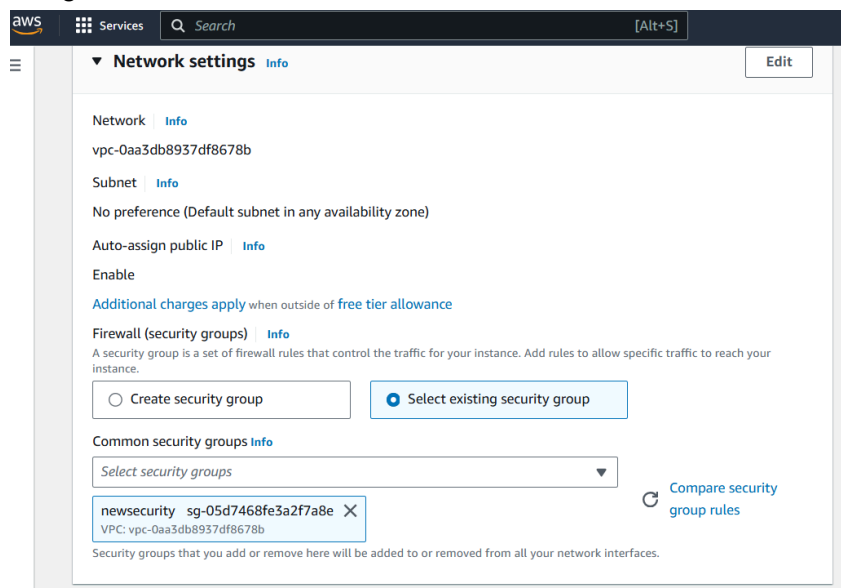
Run this command **sudo systemctl status**

```
ec2-user@ip-172-31-41-160:~/downloads/nagios-plugins-2.4.11
[ec2-user@ip-172-31-41-160 nagios-plugins-2.4.11]$ sudo systemctl status
ip-172-31-41-160.ec2.internal
State: running
Units: 296 loaded (incl. loaded aliases)
Jobs: 0 queued
Failed: 0 units
Since: Wed 2024-10-02 12:28:05 UTC; 33min ago
systemd: 252.23-2.amzn2023
CGroup: /
└─init.scope
   └─1 /usr/lib/systemd/systemd --switched-root --system --deserialize=32
      └─system.slice
         └─acpid.service
            └─1938 /usr/bin/systemd-inhibit --what=handle-suspend-key:handle-hibernate-key --who=noah "--why=acpid instead" --mode=block /usr/sbin/acpid -f
            └─2059 /usr/sbin/acpid -f
            └─amazon-ssm-agent.service
               └─7141 /usr/bin/amazon-ssm-agent
            └─atd.service
               └─2152 /usr/sbin/atd -f
            └─auditd.service
               └─1768 /sbin/auditd
            └─chronyd.service
               └─2175 /usr/sbin/chronyd -F 2
            └─dbus-broker.service
               └─1946 /usr/bin/dbus-broker-launch --scope system --audit
               └─1954 dbus-broker --log 4 --controller 9 --machine-id ec2e4d759a3e2f6fe850b14e4cdacabe --max-bytes 536870912 --max-fds 4096 --max-matches 16384 --audit
            └─gssproxy.service
               └─1959 /usr/sbin/gssproxy -D
            └─httpd.service
               └─49553 /usr/sbin/httpd -DFOREGROUND
               └─49555 /usr/sbin/httpd -DFOREGROUND
               └─49556 /usr/sbin/httpd -DFOREGROUND
               └─49557 /usr/sbin/httpd -DFOREGROUND
               └─49558 /usr/sbin/httpd -DFOREGROUND
               └─62800 /usr/sbin/httpd -DFOREGROUND
            └─libstoragemgmt.service
               └─1040 /usr/bin/lsm -d
```

Step 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 'nagios-client' alongside the host.



▼

Key pair (login)

Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

▼

[Create new key pair](#)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
nagios-host	i-03facef442a77494d	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-34-229-45-75
nagios-client	i-0b934b61f21351c1b	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-54-172-92-22

Step 3: TO BE DONE IN THE Nagios-host TERMINAL

In the nagios-host terminal, run this command

ps -ef | grep nagios

```
[ec2-user@ip-172-31-41-160 nagios-plugins-2.4.11]$ ps -ef | grep nagios
ec2-user  63115    2315    0 13:03 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-41-160 nagios-plugins-2.4.11]$
```

To become a root user, run **'sudo su'** and make two directories using the following commands. If one is running these commands in windows powershell, make sure that he/she copies it line by line as powershell might make an error while interpreting multiple lines

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

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

```
[ec2-user@ip-172-31-92-249 ~]$ sudo su
[root@ip-172-31-92-249 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-92-249 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-92-249 ec2-user]#
```

Copy the sample localhost.cfg file to linuxhost folder. Use the following mentioned command to achieve it

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

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

Open linuxserver.cfg using nano and make the following changes. This is a conf type file in which we will have to modify the configurations in way which will help us specify the hosts and clients to be monitored

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

Changes to be made:

1. Change the hostname to linux-server (EVERYWHERE ON THE FILE)
2. Change address to the public IP address of your LINUX CLIENT.
3. Change hostgroup_name under hostgroup to linux-servers1

```
# 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          linux-server
    alias              localhost
    address            54.172.92.226
}
```

```
# 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 group
}
```

IMP: Everywhere else on the file, change the hostname to linux-server instead of localhost.

Open the Nagios Config file and add the following line

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

Add the following line in the file and save

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

```
# 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

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/
# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg
```

Verify the configuration files by running the following command

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

```
[root@ip-172-31-41-160 nagios-plugins-2.4.11]# /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
[root@ip-172-31-41-160 nagios-plugins-2.4.11]#
```

You are good to go if there are no errors.

Restart the nagios service

service nagios restart

And by running `sudo systemctl status nagios`, we can again check whether our server is running or not

```

root@ip-172-31-41-160:/tmp/nagios-plugins-2.4.11
[root@ip-172-31-41-160 nagios-plugins-2.4.11]# sudo systemctl restart nagios
[root@ip-172-31-41-160 nagios-plugins-2.4.11]# 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 Wed 2024-10-02 13:20:17 UTC; 7s ago
     Docs: https://www.nagios.org/documentation
   Process: 78776 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 78777 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 78778 (nagios)
     Tasks: 6 (limit: 1112)
    Memory: 4.0M
       CPU: 24ms
   CGroup: /system.slice/nagios.service
           └─78778 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─78779 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─78780 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─78781 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─78782 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─78783 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: qh: echo service query handler registered
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: qh: help for the query handler registered
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: Successfully registered manager as @wproc with query handler
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: Registry request: name=Core Worker 78782;pid=78782
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: Registry request: name=Core Worker 78781;pid=78781
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: Registry request: name=Core Worker 78780;pid=78780
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: Registry request: name=Core Worker 78779;pid=78779
Oct 02 13:20:17 ip-172-31-41-160.ec2.internal nagios[78778]: Successfully launched command file worker with pid 78783
Oct 02 13:20:21 ip-172-31-41-160.ec2.internal nagios[78778]: HOST ALERT: linux-server;UP;SOFT;1;PING OK - Packet loss = 0%, RTA = 0.93 ms
Oct 02 13:20:24 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE ALERT: localhost;HTTP;WARNING;HARD;4;HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.0
[root@ip-172-31-41-160 nagios-plugins-2.4.11]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/httpd.service.d
            └─php-fpm.conf
   Active: active (running) since Wed 2024-10-02 12:47:56 UTC; 33min ago
     Docs: man:httpd.service(8)
   Main PID: 49553 (httpd)
    Status: "Total requests: 26; Idle/Busy workers 100/0;Requests/sec: 0.0129; Bytes served/sec: 94 B/sec"
     Tasks: 230 (limit: 1112)
    Memory: 21.7M
       CPU: 1.416s
   CGroup: /system.slice/httpd.service
           └─49553 /usr/sbin/httpd -fpm -c /etc/httpd/conf/php-fpm.conf

```

Step 4: TO BE DONE IN THE Nagios-client TERMINAL

Now it is time to switch to the client machine.

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

```

PS C:\WINDOWS\system32> cd C:\Users\Bell\Downloads
PS C:\Users\Bell\Downloads> ssh -i "mohit.pem" ubuntu@ec2-54-172-92-226.compute-1.amazonaws.com
The authenticity of host 'ec2-54-172-92-226.compute-1.amazonaws.com (54.172.92.226)' can't be established.
ECDSA key fingerprint is SHA256:e/WkFQRuHSqPjqQ5hDMA0dku8msNHEtN9SAgzEy53E.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-172-92-226.compute-1.amazonaws.com,54.172.92.226' (ECDSA) 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 Wed Oct  2 13:26:11 UTC 2024

System load:  0.0          Processes:      104
Usage of /:   22.8% of 6.71GB Users logged in:   0
Memory usage: 20%         IPv4 address for enx0: 172.31.36.100
Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

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 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

```

Make a package index update and install gcc, nagios-nrpe-server and the plugins. Run the following commands to achieve the same.

sudo apt update -y

sudo apt install gcc -y

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

[illegible]

```

ubuntu@ip-172-31-36-100:~$ sudo apt install -y nagios-nrpe-server nagios-plugins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'monitoring-plugins' instead of 'nagios-plugins'
The following additional packages will be installed:
libavahi-client3 libavahi-common-data libavahi-common3 libbz2-dev libcbor2t64 libcurl4 libdbus-1-3 libedit2 libevent-2.1-7 libexpat1-dev libffi8 libfribidi0 libgdbm-compat4 libgnutls30 libhogweed4 libidn2-0 libldap2 libltdl7 liblua5.4-0 libnet-snmp-perl libnpptools libnspr4 libnss3 libodbc1 libopenldap2 liborc-0.4-0 libpython-gpg python3-kamion python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common samba-common-bin samba-dfs-modules samba-lsmbc smbclient snmp
Suggested packages:
 cups-common libcrypt-deb-perl libdixie-hmac-perl libio-socket-inet-perl perl-smms-downloader icinga2 nagios-plugins-contrib fping postfix | sendmail-bin | exim4-daemon-heavy
libavahi-client3 libidn2-0 libldap2 libltdl7 liblua5.4-0 libnet-snmp-perl libnpptools libnspr4 libnss3 libodbc1 libopenldap2 liborc-0.4-0 libpython-gpg python3-kamion python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common samba-common-bin samba-dfs-modules samba-lsmbc smbclient snmp
The following NEW packages will be installed:
libavahi-client3 libavahi-common-data libavahi-common3 libbz2-dev libcbor2t64 libcurl4 libdbus-1-3 libedit2 libevent-2.1-7 libexpat1-dev libffi8 libfribidi0 libgdbm-compat4 libgnutls30 libhogweed4 libidn2-0 libldap2 libltdl7 liblua5.4-0 libnet-snmp-perl libnpptools libnspr4 libnss3 libodbc1 libopenldap2 liborc-0.4-0 libpython-gpg python3-kamion python3-markdown python3-samba python3-talloc python3-tdb rpcbind samba-common samba-common-bin samba-dfs-modules samba-lsmbc smbclient snmp
1 upgraded, 39 newly installed, 0 to remove and 6 not upgraded.
Need to get 16.1 MB of archives.
After this operation, 72.9 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 nagios-nrpe-server amd64 4.1.0-ubuntu2 [356 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 rpcbind amd64 1.2.6-ubuntu2 [46.5 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libavahi-common-data libavahi-common-data 0.8-1ubuntu2 [29.7 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libavahi-client3 libavahi-client3 0.8-1ubuntu2 [29.3 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libavahi-common3 libavahi-common3 0.8-1ubuntu2 [29.3 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main amd64 libcbor2t64 amd64 2.4.7.1-ubuntu2.3 [272 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libidn2-0 amd64 0.9-0.6-build1 [25.7 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 librtldbd amd64 2.4.2-1build2 [27.3 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libtdb1 amd64 1.4.10-1build1 [46.8 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libevent0t64 amd64 0.9.11-2build1 [42.6 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libldb2 amd64 2:2.8.0+samba4-19.5dfsg-ubuntu2 [187 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 mysql-common all 5.8.01-1build1 [6746 B]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main amd64 libmysqldclient21 amd64 0.8.39-0ubuntu2.04.02 [1294 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libnet-snmp-perl all 6.0.1-7 [86.8 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble-updates/main amd64 libnpptools amd64 16.4-ubuntu2.04.02 [74.6 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 libradcli1 amd64 1.2-11build1 [40.5 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libwbclient0 amd64 2:19.5dfsg-ubuntu2 [191 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 samba-lsmbc amd64 2.19.5dfsg-ubuntu2 [6017 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libsnmp-base amd64 2:19.5dfsg-ubuntu2 [62.4 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libsnmp-perl all 5.9.4dfsg-1ubuntu2 [206 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 libsnmpdot4t64 amd64 5.9.4dfsg-1ubuntu2 [1066 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 libripasert0 amd64 0.9.7dfsg-2build1 [35.8 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 python3-gpg amd64 1.8.0-0ubuntu2 [209 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 python3-libidn2 amd64 2:2.8.0+samba4-19.5dfsg-ubuntu2 [41.8 kB]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu/noble/main amd64 python3-markdown all 3.5.2-1 [72.0 kB]

```

Open nrpe.cfg file to make changes.

sudo nano /etc/nagios/nrpe.cfg

Under allowed_hosts, add your nagios host IP address like so

```
ubuntu@ip-172-31-36-100: ~
GNU nano 7.2
#
# 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=127.0.0.1,34.229.45.75
#
# 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
```

Now restart the NRPE server by this command.

sudo systemctl restart nagios-nrpe-server

```
ubuntu@ip-172-31-36-100: ~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-36-100: ~$
```

Run the following command in the Nagios-host terminal

sudo systemctl status nagios

```
[root@ip-172-31-41-160 nagios-plugins-2.4.11]# sudo systemctl status nagios
● nagios.service - nagios core 4.5.3
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Wed 2024-10-02 13:20:17 UTC; 15min ago
     Docs: https://www.nagios.org/documentation
   Main PID: 78778 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 4.3M
      CPU: 403ms
  CGroup: /system.slice/nagios.service
          └─78778 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─78779 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                └─78780 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─78781 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                      └─78782 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                         └─78783 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE NOTIFICATION: nagiosadmin;localhost;Swap Usage;CRITICAL;notify-service-by-email;SWAP CRITICAL - 0% free (0 MB out of 0
Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: NOTIFY job 3 from worker core Worker 78782 is a non-check helper but exited with return code 127
Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: host=localhost; service=Swap Usage; contact=nagiosadmin
Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: early timeout=0; exited ok=1; wait status=32512; error code=0;
Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: stderr line 01: /bin/sh; line 1: /bin/mail: No such file or directory
Oct 02 13:22:54 ip-172-31-41-160.ec2.internal nagios[78778]: wproc: stderr line 02: /usr/bin/printf: write error: Broken pipe
Oct 02 13:23:13 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE ALERT: linux-server;Total Processes;OK;HARD;1;PROCS OK: 37 processes with STATE = RSZDT
Oct 02 13:23:50 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE ALERT: linux-server;Current Load;OK;HARD;1;OK - load average: 0.01, 0.07, 0.04
Oct 02 13:24:28 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE ALERT: linux-server;Current Users;OK;HARD;1;USERS OK - 2 users currently logged in
Oct 02 13:24:46 ip-172-31-41-160.ec2.internal nagios[78778]: SERVICE ALERT: localhost;Current Users;OK;HARD;1;USERS OK - 2 users currently logged in
lines 1-26/26 (END)
```

Step 5: Visiting your nagios server using your nagios-host ip address

Open up your browser and look for `http://<public_ip_address_of_nagios-host>/nagios`

The screenshot shows the Nagios Core web interface. The browser address bar displays `34.229.45.75/nagios/`. The page features a left sidebar with navigation links under categories: General (Home, Documentation), Current Status (Tactical Overview, Map, Hosts, Services, Host Groups, Service Groups, Problems, Quick Search), and Reports (Availability, Trends, Alerts, History, Summary, Histogram, Notifications, Event Log). The main content area displays the Nagios Core logo, a green checkmark indicating the daemon is running with PID 78778, and the version 4.5.5 as of September 17, 2024. Below this, there are sections for 'Get Started' (with links like 'Start monitoring your infrastructure'), 'Quick Links' (with links like 'Nagios Library'), 'Latest News', and 'Don't Miss...'. A 'Page Tour' button is visible in the bottom right corner.

Click on Hosts.

The screenshot shows the Nagios Core web interface with the 'Hosts' page selected. The browser address bar displays `34.229.45.75/nagios/`. The left sidebar is the same as in the previous screenshot. The main content area displays the 'Current Network Status' (Last Updated: Wed Oct 2 13:40:35 UTC 2024) and 'Host Status Totals' (Up: 2, Down: 0, Unreachable: 0, Pending: 0). Below this, there are 'Service Status Totals' (Ok: 12, Warning: 1, Unknown: 0, Critical: 3, Pending: 0) and a table titled 'Host Status Details For All Host Groups'. The table has columns: Host, Status, Last Check, Duration, and Status Information. It shows two hosts: 'linux-server' and 'localhost', both with a status of 'UP'. A 'Limit Results' dropdown is set to '100'. A 'Page Tour' button is visible in the bottom right corner.

Host	Status	Last Check	Duration	Status Information
linux-server	UP	10-02-2024 13:40:17	0d 0h 20m 18s	PING OK - Packet loss = 0%, RTA = 0.84 ms
localhost	UP	10-02-2024 13:40:09	0d 0h 20m 26s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Click on linux-server to view host information

The screenshot displays the Nagios web interface for the host 'linux-server'. The interface includes a sidebar with navigation links such as General, Current Status, Tactical Overview, Map, Hosts, Services, Host Groups, Service Groups, Problems, Reports, and System. The main content area is divided into several sections:

- Host Information:** Shows the host is 'UP' (for 0d 0h 20m 39s), with status information, performance data, current attempt, last check time, check type, check latency, next scheduled active check, last state change, last notification, and scheduled downtime.
- Host State Information:** A summary of the host's state, including active checks, passive checks, obsessing, notifications, event handler, and flap detection.
- Host Commands:** A list of commands that can be executed on the host, such as 'Locate host on map', 'Disable active checks of this host', 'Re-schedule the next check of this host', etc.
- Host Comments:** A section for adding or deleting comments about the host.

The bottom of the interface shows a Windows taskbar with the date and time (19:11, 02-10-2024) and system status (31°C, Haze).

We can even navigate to the services section, which explicitly mentions the status, duration, checks, information about the numerous services present on our hosts

The screenshot displays the Nagios web interface for the 'Service Status Details For All Hosts' section. The interface includes a sidebar with navigation links such as General, Current Status, Tactical Overview, Map, Hosts, Services, Host Groups, Service Groups, Problems, Reports, and System. The main content area is divided into several sections:

- Current Network Status:** Shows the current network status, including the last updated time, update frequency, and Nagios version.
- Host Status Totals:** A summary of the host's status, including up, down, unreachable, and pending states.
- Service Status Totals:** A summary of the service's status, including ok, warning, unknown, critical, and pending states.
- Service Status Details For All Hosts:** A table showing the status of various services across different hosts. The table includes columns for Host, Service, Status, Last Check, Duration, Attempt, and Status Information.

The bottom of the interface shows a Windows taskbar with the date and time (19:11, 02-10-2024) and system status (31°C, Haze).

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.