

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:

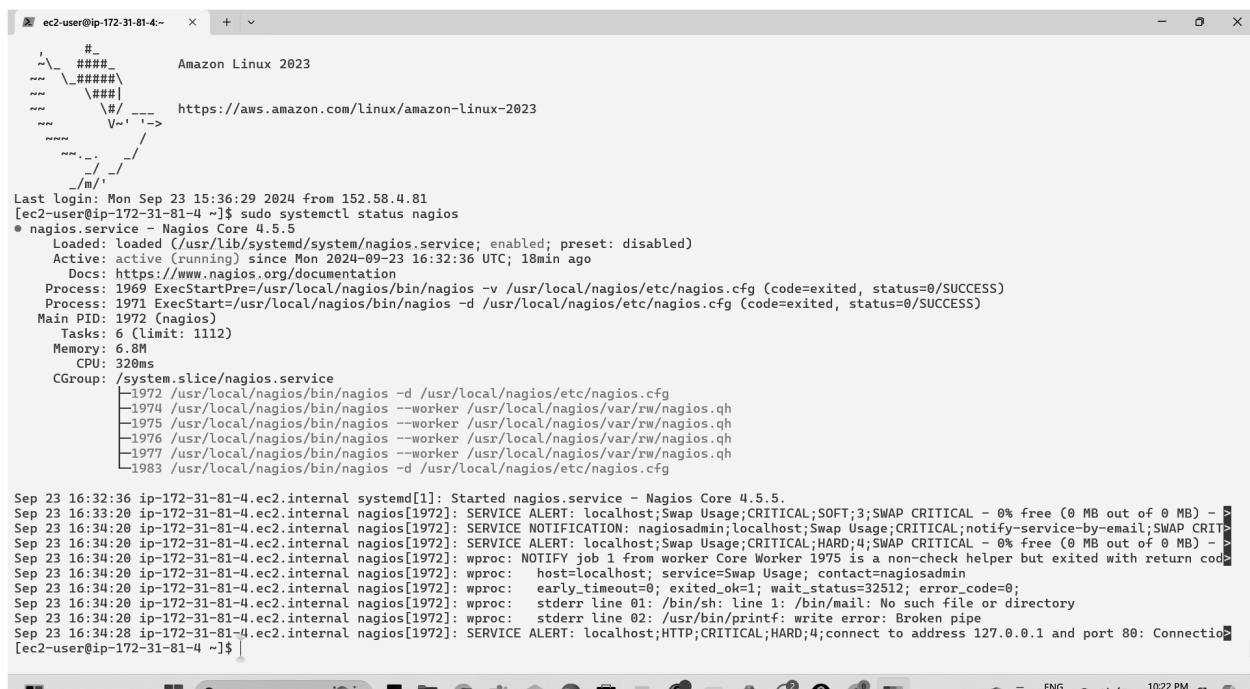
AWS Academy or Personal account.

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



```
ec2-user@ip-172-31-81-4:~$ sudo systemctl status nagios
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Mon Sep 23 15:36:29 2024 from 152.58.4.81
[ec2-user@ip-172-31-81-4 ~]$ 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 Mon 2024-09-23 16:32:36 UTC; 18min ago
     Docs: https://www.nagios.org/documentation
    Process: 1969 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
    Process: 1971 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 1972 (nagios)
      Tasks: 6 (Limit: 1112)
     Memory: 6.8M
        CPU: 320ms
    CGroup: /system.slice/nagios.service
            └─1972 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              └─1974 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                └─1975 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                  └─1976 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                    └─1977 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                      └─1983 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 23 16:32:36 ip-172-31-81-4.ec2.internal systemd[1]: Started nagios.service - Nagios Core 4.5.5.
Sep 23 16:33:20 ip-172-31-81-4.ec2.internal nagios[1972]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;SOFT;3;SWAP CRITICAL - 0% free (0 MB out of 0 MB) - >
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: SERVICE NOTIFICATION: nagiosadmin;localhost;Swap Usage;CRITICAL;notify-service-by-email;SWAP CRIT
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: SERVICE ALERT: localhost;Swap Usage;CRITICAL;HARD;4;SWAP CRITICAL - 0% free (0 MB out of 0 MB) - >
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: wproc: NOTIFY job 1 from worker Core Worker 1975 is a non-check helper but exited with return cod
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: wproc: host=localhost; service=Swap Usage; contact=nagiosadmin
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: wproc: early_timeout=0; exited_ok=1; wait_status=32512; error_code=0;
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: wproc: stderr line 01: /bin/sh: line 1: /bin/mail: No such file or directory
Sep 23 16:34:20 ip-172-31-81-4.ec2.internal nagios[1972]: wproc: stderr line 02: /usr/bin/printf: write error: Broken pipe
Sep 23 16:34:28 ip-172-31-81-4.ec2.internal nagios[1972]: SERVICE ALERT: localhost;HTTP;CRITICAL;HARD;4;connect to address 127.0.0.1 and port 80: Connection
[ec2-user@ip-172-31-81-4 ~]$
```

Name: Bhushan Mukund Kor

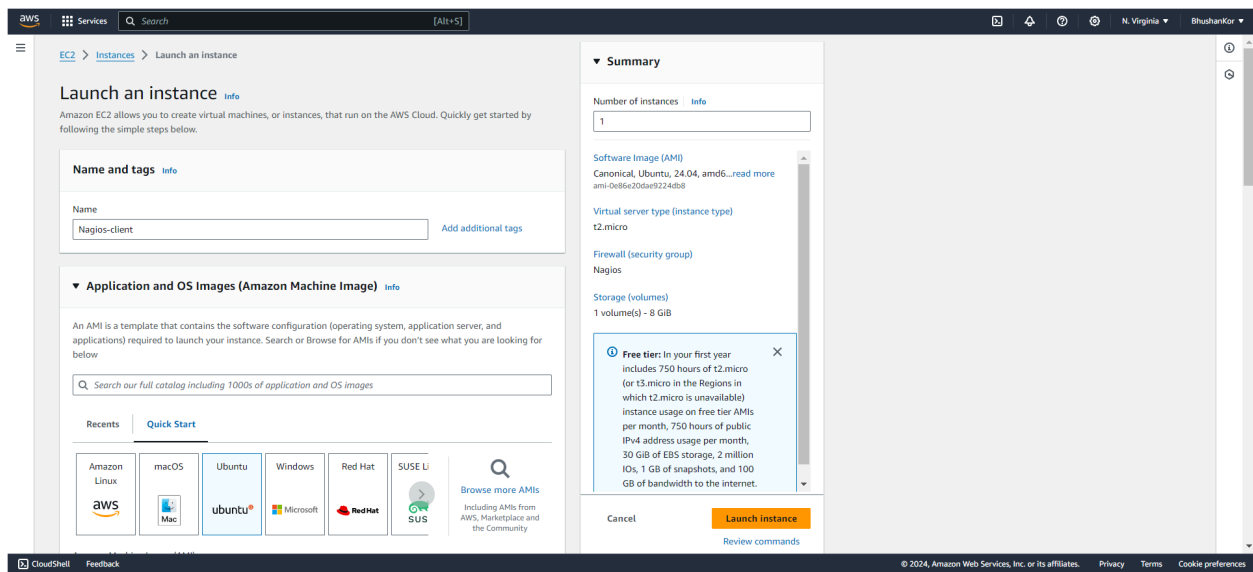
Academic Year: 2024-2025

Division: D15C

Roll No: 28

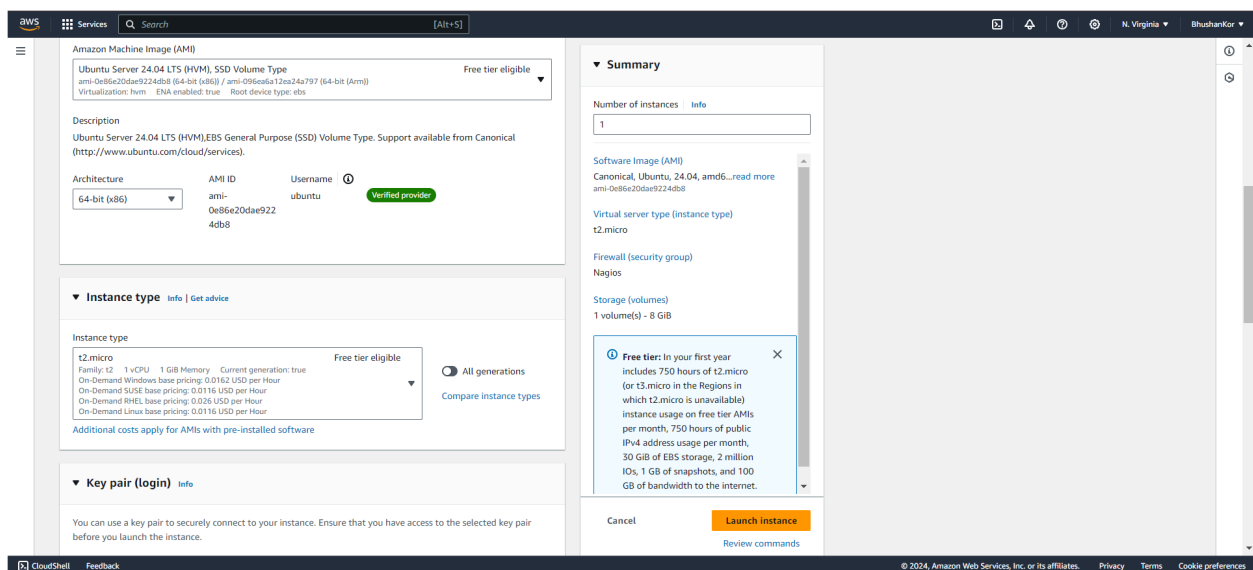
You can now proceed if you get the above message/output.

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

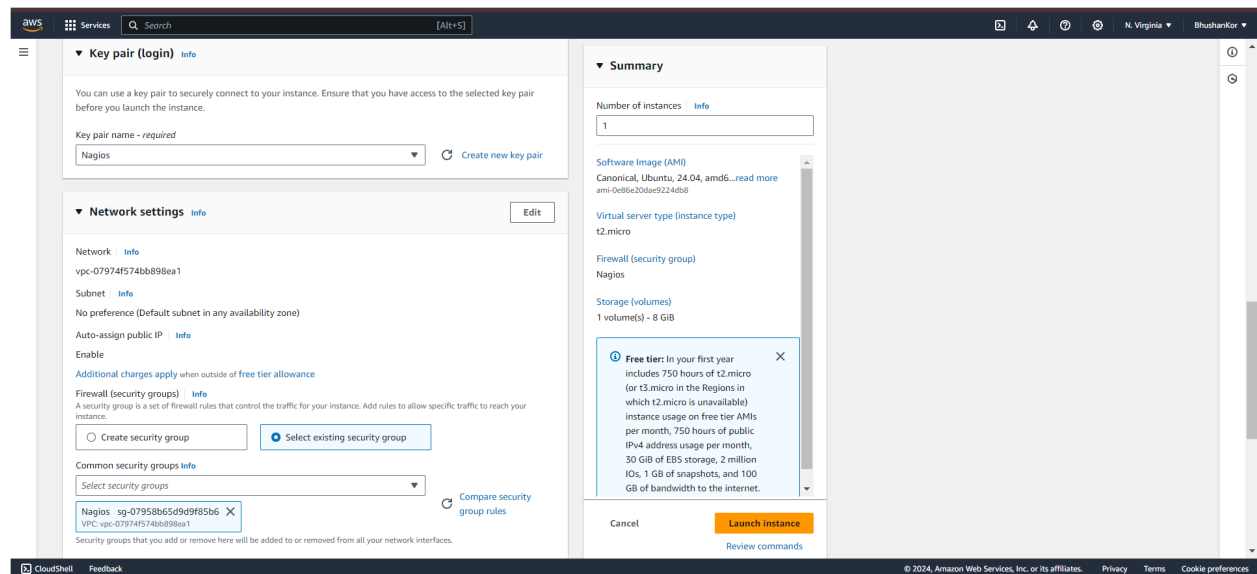


For Key pair : Click on create key and make key of type RSA with extension .pem . Key will be downloaded to your local machine.

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.

```

ubuntu@ip-172-31-83-152: ~
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\bhush\one drive 2\OneDrive\Desktop\New folder (5)> ssh -i "Nagios.pem" ubuntu@ec2-3-88-57-181.compute-1.amazonaws.com
The authenticity of host 'ec2-3-88-57-181.compute-1.amazonaws.com (64:ff9b::358:39b5)' can't be established.
ED25519 key fingerprint is SHA256:r48lMslkuQqtE3zFw3vqd7aKVfOk5+Kzkh1k6l3cDA4.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-88-57-181.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Sep 23 16:55:50 UTC 2024

System load:  0.81          Processes:           119
Usage of /:   22.8% of 6.71GB Users logged in:      0
Memory usage: 21%          IPv4 address for enX0: 172.31.83.152
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

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
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-83-152:~$

```

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-172-31-81-4 ~]$ ps -ef | grep nagios
nagios      1972      1  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios      1974    1972  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      1975    1972  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      1976    1972  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      1977    1972  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios      1983    1972  0 16:32 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user    3135    2898  0 16:57 pts/0    00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-81-4 ~]$

```

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

```

[root@ip-172-31-81-4 ec2-user]# sudo su

```

```

[root@ip-172-31-81-4 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-81-4 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-81-4 ec2-user]#

```

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-172-31-81-4 ec2-user]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-81-4 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

```
GNU nano 5.8 /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg Modified

#####
#
# 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            172.31.83.152
}

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

# Define an optional hostgroup for Linux machines

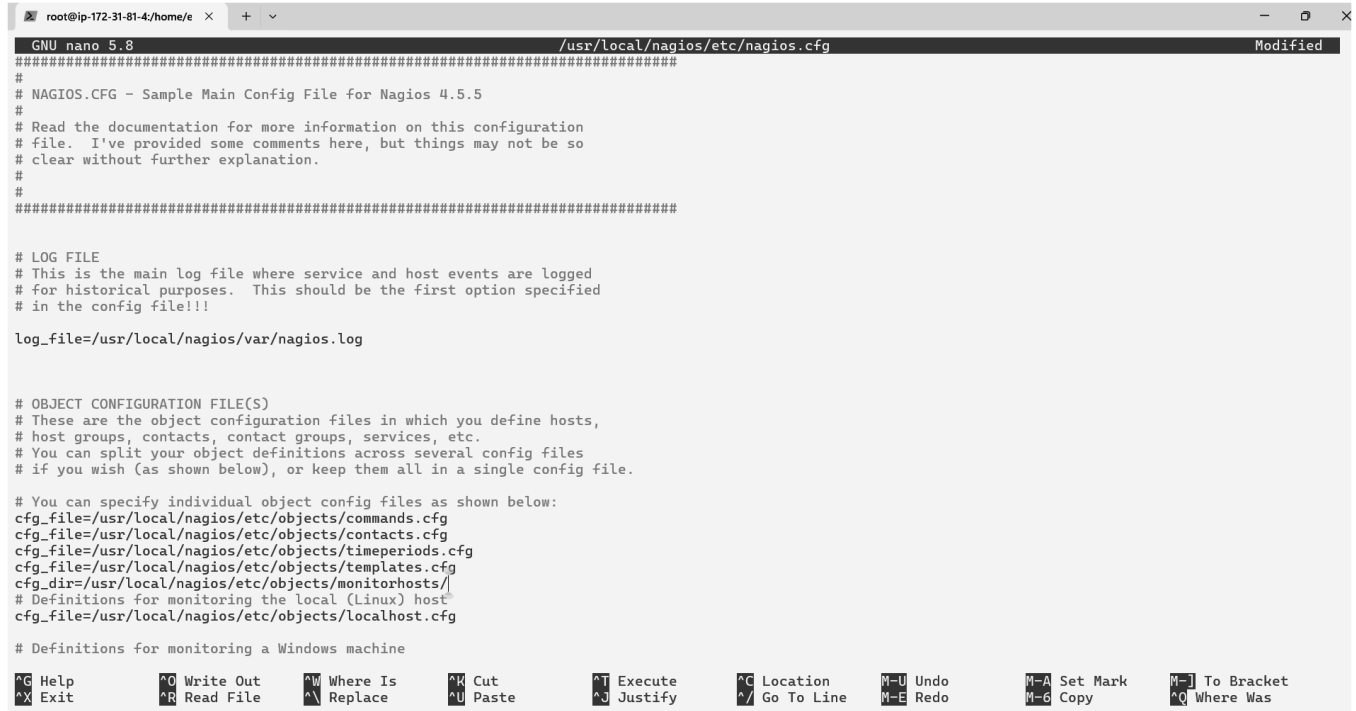
define hostgroup {
    hostgroup_name      linux-servers1              ; The name of the hostgroup
    alias               Linux Servers                ; Long name of the group
    members             localhost                    ; Comma separated list of hosts that belong to this group
}

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C Location  M-U Undo     M-A Set Mark  M-] To Bracket
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify   ^_ Go To Line M-E Redo     M-G Copy      ^Q Where Was
```

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`



```

GNU nano 5.8 /usr/local/nagios/etc/nagios.cfg Modified
#####
# NAGIOS.CFG - Sample Main Config File for Nagios 4.5.5
#
# 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!!!

log_file=/usr/local/nagios/var/nagios.log

# 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/
# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^U Undo       ^A Set Mark   ^_ To Bracket
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^_/ Go To Line ^E Redo       ^G Copy       ^Q Where Was

```

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-172-31-81-4 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
[root@ip-172-31-81-4 ec2-user]#

```

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

```
[root@ip-172-31-81-4 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-81-4 ec2-user]# sudo systemctl status nagios
```

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-172-31-83-152:~$ 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
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
135 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu gcc-13 gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libasan8 libatomic1 libcc1-0
  libgcc-13-dev libgomp1 libhwasa0 libisl23 libitm1 liblsan0 libmpc3 libquadmath0 libtsan2 libubsan1
Suggested packages:
  cpp-doc gcc-13-localles cpp-13-doc gcc-multilib make autoconf automake libtool flex bison gdb gcc-doc gcc-13-multilib gcc-13-doc gdb-x86-64-linux-gnu
The following NEW packages will be installed:
  cpp cpp-13 cpp-13-x86-64-linux-gnu cpp-x86-64-linux-gnu gcc gcc-13 gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libasan8 libatomic1 libcc1-0
  libgcc-13-dev libgomp1 libhwasa0 libisl23 libitm1 liblsan0 libmpc3 libquadmath0 libtsan2 libubsan1
0 upgraded, 22 newly installed, 0 to remove and 135 not upgraded.
Need to get 47.3 MB of archives.
After this operation, 163 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 gcc-13-base amd64 13.2.0-23ubuntu4 [49.0 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libisl23 amd64 0.26-3build1 [680 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libmpc3 amd64 1.3.1-1build1 [54.5 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 cpp-13-x86-64-linux-gnu amd64 13.2.0-23ubuntu4 [11.2 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 cpp-13 amd64 13.2.0-23ubuntu4 [1032 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 cpp-x86-64-linux-gnu amd64 4:13.2.0-7ubuntu1 [5326 B]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 cpp amd64 4:13.2.0-7ubuntu1 [22.4 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libcc1-0 amd64 14-20240412-0ubuntu1 [47.7 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libgomp1 amd64 14-20240412-0ubuntu1 [147.7 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libitm1 amd64 14-20240412-0ubuntu1 [28.9 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libatomic1 amd64 14-20240412-0ubuntu1 [10.4 kB]
```

```
ubuntu@ip-172-31-83-152:~$ sudo apt install -y nagios-nrpe-server nagios-plugins
monitoring-plugins is already the newest version (2.3.5-1ubuntu3).
Suggested packages:
  xinetd | inetd
The following NEW packages will be installed:
  nagios-nrpe-server
0 upgraded, 1 newly installed, 0 to remove and 135 not upgraded.
Need to get 356 kB of archives.
After this operation, 469 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 nagios-nrpe-server amd64 4.1.0-1ubuntu3 [356 kB]
Fetched 356 kB in 0s (15.6 MB/s)
Selecting previously unselected package nagios-nrpe-server.
(Reading database ... 73771 files and directories currently installed.)
Preparing to unpack .../nagios-nrpe-server_4.1.0-1ubuntu3_amd64.deb ...
Unpacking nagios-nrpe-server (4.1.0-1ubuntu3) ...
Setting up nagios-nrpe-server (4.1.0-1ubuntu3) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...

Service restarts being deferred:
/etc/needrestart/restart.d/dbus.service
systemctl restart getty@tty1.service
systemctl restart networkd-dispatcher.service
systemctl restart serial-getty@ttyS0.service
systemctl restart systemd-logind.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
ubuntu @ session #2: sshd[1001,1111]
ubuntu @ session #8: sshd[1205,1263]
ubuntu @ user manager service: systemd[1006]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-83-152:~$
```

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
# This determines the effective user that the NRPE daemon should run as.
# You can either supply a username or a UID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd
nrpe_user=nagios

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

# 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,::1,3.86.12.126

# COMMAND ARGUMENT PROCESSING
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^U Undo       ^M Set Mark   ^] To Bracket
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line  ^E Redo       ^G Copy       ^_ Where Was

```

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

```

ubuntu@ip-172-31-83-152:~$ sudo nano /etc/nagios/nrpe.cfg

ubuntu@ip-172-31-83-152:~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-83-152:~$

```

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

```

[root@ip-172-31-81-4 ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@ip-172-31-81-4 ec2-user]# 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 Mon 2024-09-23 17:26:10 UTC; 1min 39s ago
     Docs: https://www.nagios.org/documentation
   Process: 4227 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 4228 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 4234 (nagios)
    Tasks: 6 (limit: 1112)
   Memory: 4.2M
     CPU: 60ms
   CGroup: /system.slice/nagios.service
           └─4234 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─4236 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─4237 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                 └─4238 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─4239 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                     └─4242 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: qh: echo service query handler registered
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: qh: help for the query handler registered
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: wproc: Successfully registered manager as @wproc with query handler
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: wproc: Registry request: name=Core Worker 4238;pid=4238
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: wproc: Registry request: name=Core Worker 4239;pid=4239
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: wproc: Registry request: name=Core Worker 4237;pid=4237
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: wproc: Registry request: name=Core Worker 4236;pid=4236
Sep 23 17:26:10 ip-172-31-81-4.ec2.internal nagios[4234]: Successfully launched command file worker with pid 4242
Sep 23 17:27:00 ip-172-31-81-4.ec2.internal nagios[4234]: SERVICE ALERT: linuxserver;HTTP;CRITICAL;SOFT;3;connect to address 172.31.83.152 and port 80: Con
Sep 23 17:27:30 ip-172-31-81-4.ec2.internal nagios[4234]: SERVICE ALERT: linuxserver;Swap Usage;CRITICAL;SOFT;1;SWAP CRITICAL - 0% free (0 MB out of 0 MB)
[root@ip-172-31-81-4 ec2-user]#

```



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

```
Step 23: [root@ip-172-31-81-4 ec2-user]# sudo systemctl status httpd
[root@ip-172-31-81-4 ec2-user]# 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: inactive (dead)
   Docs: man:httpd.service(8)
[root@ip-172-31-81-4 ec2-user]# sudo systemctl start httpd
[root@ip-172-31-81-4 ec2-user]# sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-81-4 ec2-user]#
```

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

Nagios® Core™
✓ Daemon running with PID 4835

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 or unregistered trademarks owned by Nagios Enterprises, LLC. None of the Nagios marks is assumed to be associated with this trademark.com certification.

Now Click on Hosts from left side panel

Nagios®

Current Network Status
Last Updated: Mon Sep 23 17:50:58 UTC 2024
Updated every 30 seconds
Nagios® Core™ 4.5.5 - www.nagios.org
Logged in as nagiosadmin

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
12	1	0	3	0

Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-23-2024 17:47:30	0d 0h 27m 50s	PING OK - Packet loss = 0%, RTA = 0.03 ms
localhost	UP	09-23-2024 17:49:28	0d 1h 56m 4s	PING OK - Packet loss = 0%, RTA = 0.04 ms

Results 1 - 2 of 2 Matching Hosts

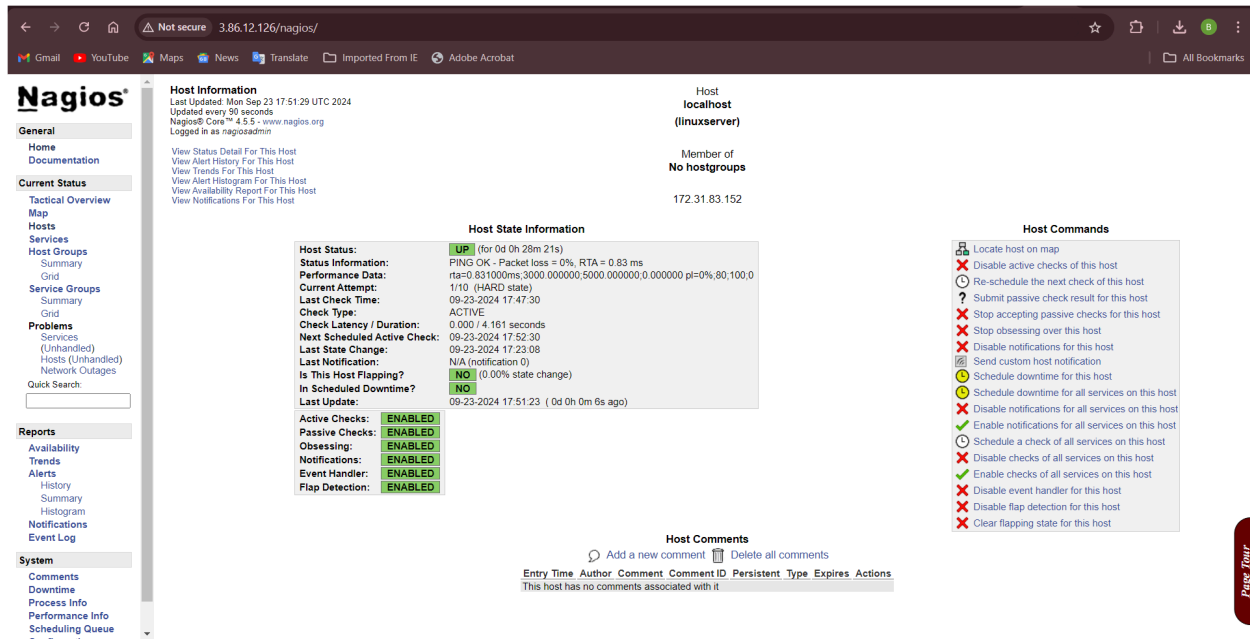
Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

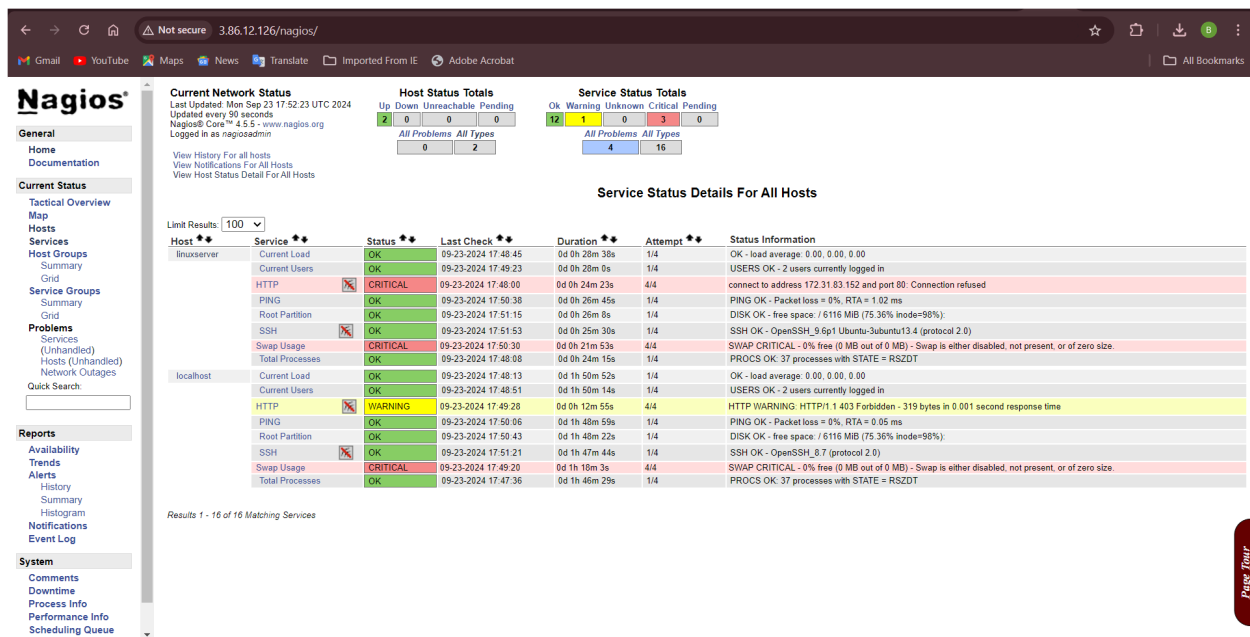
Roll No: 28

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



The screenshot displays the Nagios web interface for the host 'localhost (linuxserver)'. The interface includes a sidebar with navigation links such as 'General', 'Current Status', 'Problems', 'Reports', and 'System'. The main content area shows 'Host Information' with details like 'Last Updated: Mon Sep 23 17:51:29 UTC 2024' and 'Nagios® Core™ 4.5.5'. Below this, the 'Host State Information' section provides a comprehensive overview of the host's status, including 'Status Information' (UP), 'Performance Data' (rtt=0.831000ms), 'Current Attempt' (1/10), 'Check Type' (ACTIVE), 'Check Latency / Duration' (0.000 / 4.161 seconds), 'Next Scheduled Active Check' (09-23-2024 17:52:30), 'Last State Change' (09-23-2024 17:23:08), 'Last Notification' (N/A), 'Is This Host Flapping?' (NO), 'In Scheduled Downtime?' (NO), and 'Last Update' (09-23-2024 17:51:23). The 'Host Commands' section lists various actions like 'Locate host on map', 'Disable active checks of this host', and 'Re-schedule the next check of this host'. The 'Host Comments' section at the bottom indicates that there are no comments associated with this host.

Current Network Status



The screenshot displays the Nagios web interface showing the 'Current Network Status'. The interface includes a sidebar with navigation links such as 'General', 'Current Status', 'Problems', 'Reports', and 'System'. The main content area shows the 'Current Network Status' section with a 'Limit Results' dropdown set to 100. Below this, the 'Host Status Totals' and 'Service Status Totals' are displayed. The 'Host Status Totals' section shows 2 Up, 0 Down, 0 Unreachable, and 0 Pending. The 'Service Status Totals' section shows 12 Ok, 1 Warning, 0 Unknown, 3 Critical, and 0 Pending. The 'Service Status Details For All Hosts' section provides a detailed view of the status of various services across different hosts. The table below summarizes the status of services for 'linuxserver' and 'localhost'.

Host	Service	Status	Last Check	Duration	Attempt	Status Information	
linuxserver	Current Load	OK	09-23-2024 17:49:45	0d 0h 28m 38s	1/4	OK - load average: 0.00, 0.00, 0.00	
	Current Users	OK	09-23-2024 17:49:23	0d 0h 28m 0s	1/4	USERS OK - 2 users currently logged in	
	HTTP	CRITICAL	09-23-2024 17:48:00	0d 0h 24m 23s	4/4	connect to address 172.31.83.152 and port 80: Connection refused	
	PING	OK	09-23-2024 17:50:38	0d 0h 26m 45s	1/4	PING OK - Packet loss = 0%, RTA = 1.02 ms	
	Root Partition	OK	09-23-2024 17:51:15	0d 0h 26m 8s	1/4	DISK OK - free space / 6116 MB (75.36% inode=98%):	
	SSH	OK	09-23-2024 17:51:53	0d 0h 25m 30s	1/4	SSH OK - OpenSSH_9.6p1 Ubuntu-3ubuntu13.4 (protocol 2.0)	
	Swap Usage	CRITICAL	09-23-2024 17:50:30	0d 0h 21m 53s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.	
	Total Processes	OK	09-23-2024 17:48:08	0d 0h 24m 15s	1/4	PROCS OK: 37 processes with STATE = RSZDT	
	localhost	Current Load	OK	09-23-2024 17:48:13	0d 1h 50m 52s	1/4	OK - load average: 0.00, 0.00, 0.00
		Current Users	OK	09-23-2024 17:48:51	0d 1h 50m 14s	1/4	USERS OK - 2 users currently logged in
HTTP		WARNING	09-23-2024 17:49:20	0d 0h 12m 55s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time	
PING		OK	09-23-2024 17:50:06	0d 1h 48m 59s	1/4	PING OK - Packet loss = 0%, RTA = 0.05 ms	
Root Partition		OK	09-23-2024 17:50:43	0d 1h 48m 22s	1/4	DISK OK - free space / 6116 MB (75.36% inode=98%):	
SSH		OK	09-23-2024 17:51:21	0d 1h 47m 44s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)	
Swap Usage		CRITICAL	09-23-2024 17:49:20	0d 1h 16m 3s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.	

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.