Name- Mikil. Lalwani D15B/37

Advance DevOps lab

Experiment 10.

Aim-

To perform port, service monitoring and linux server monitoring using Nagios.

Theory-

Nagios is an open source monitoring system.

Nagios core-

This is freely available open-source monitoring software for IT products. Core contains a wide array of infrastructure monitoring through allowing plug-ins to extend its monitoring capabilities. It is base for paid Nagios monitoring system.

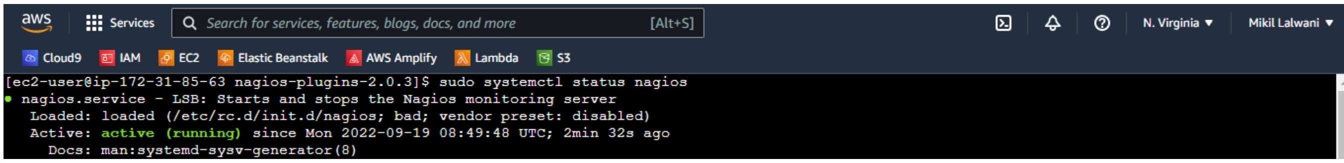
Nagios core has an optional web-interface, which displays network status, notifications, log files and more. Core can notify users when there are host or server issues.

Nagios XI is an extended version of Nagios Core, intended as an ~~server~~enterprise level version of monitoring tool. XI is added atop core with additional features and is payed to use.

Steps:

Prerequisites: AWS Free Tier, Nagios Server running on Amazon Linux Machine.

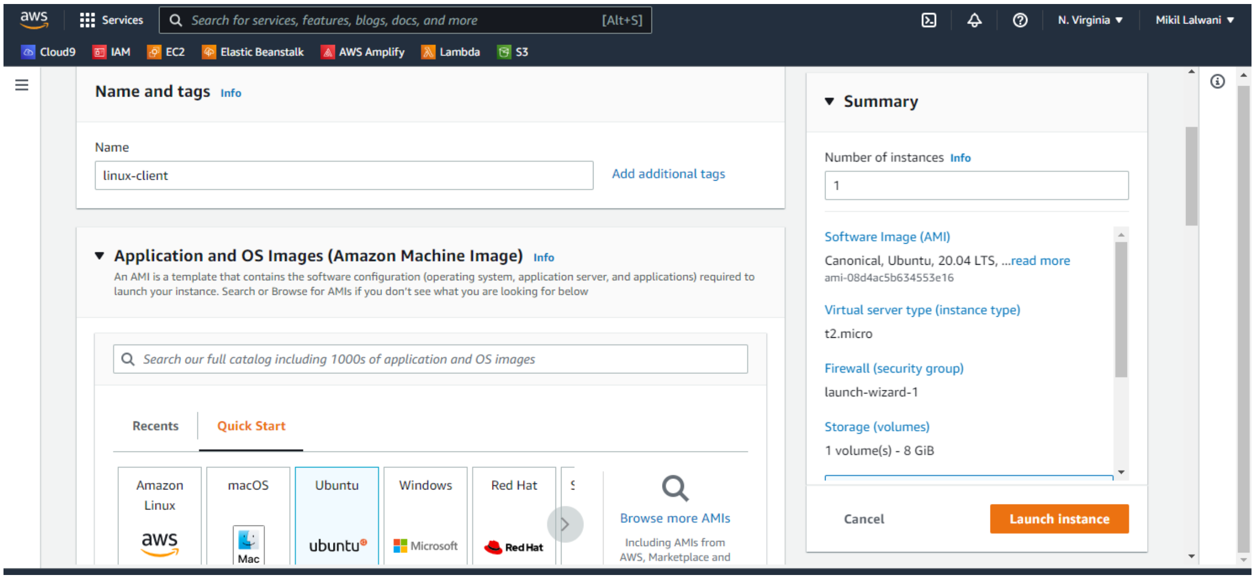
1. To Confirm that Nagios is running on the server side, run this sudo systemctl status nagios on the “NAGIOS HOST”.



```
[ec2-user@ip-172-31-85-63 nagios-plugins-2.0.3]$ sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios; bad; vendor preset: disabled)
   Active: active (running) since Mon 2022-09-19 08:49:48 UTC; 2min 32s ago
     Docs: man:systemd-sysv-generator(8)
```

   You can proceed if you get this message.

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 ‘linux-client’ alongside the host.
   For now, leave this machine as is, and go back to your nagios HOST machine.



Name and tags

Name: linux-client

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

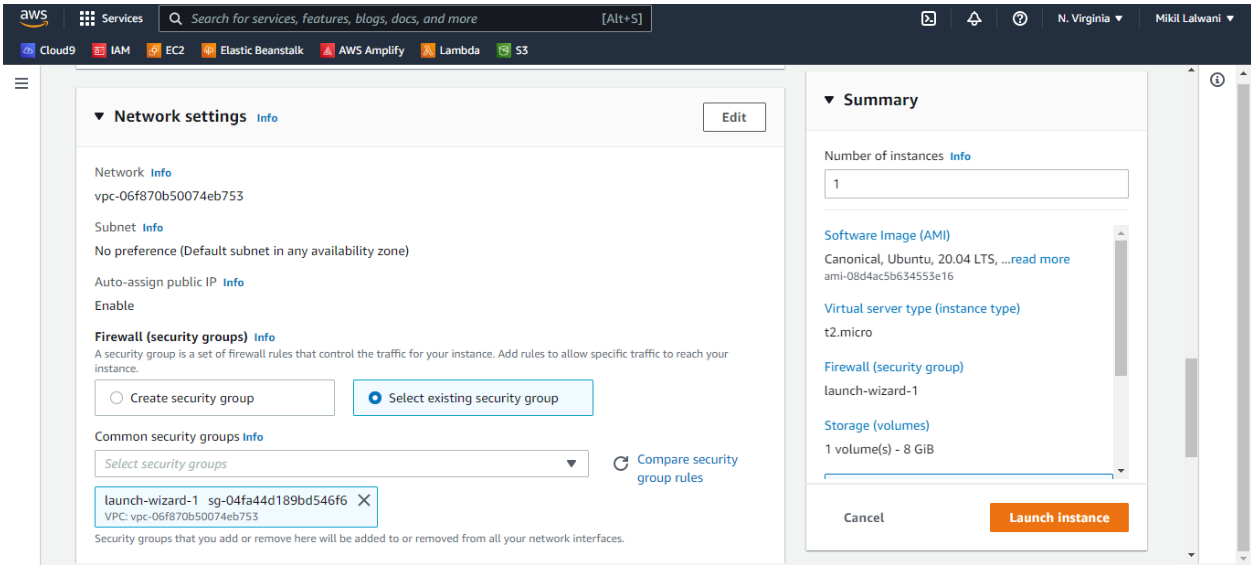
Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 20.04 LTS, ...read more
ami-08d4ac5b634553e16

Virtual server type (instance type): t2.micro

Firewall (security group): launch-wizard-1

Storage (volumes): 1 volume(s) - 8 GiB



aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
Mikil Lalwani ▼
Cloud9 IAM EC2 Elastic Beanstalk AWS Amplify Lambda S3

▼ **Network settings** Info

Network Info
vpc-06f870b50074eb753

Subnet Info
No preference (Default subnet in any availability zone)

Auto-assign public IP Info
Enable

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

○ Create security group ● Select existing security group

Common security groups Info
Select security groups ▼ Compare security group rules

launch-wizard-1 sg-04fa44d189bd546f6 ✕
VPC: vpc-06f870b50074eb753

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ **Summary**

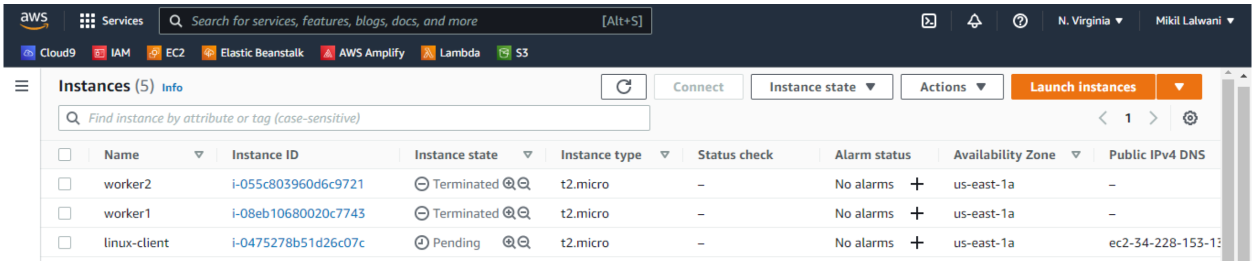
Number of instances Info
1

Software Image (AMI)
Canonical, Ubuntu, 20.04 LTS, ...read more
ami-08d4ac5b634553e16

Virtual server type (instance type)
t2.micro

Firewall (security group)
launch-wizard-1

Storage (volumes)
1 volume(s) - 8 GiB

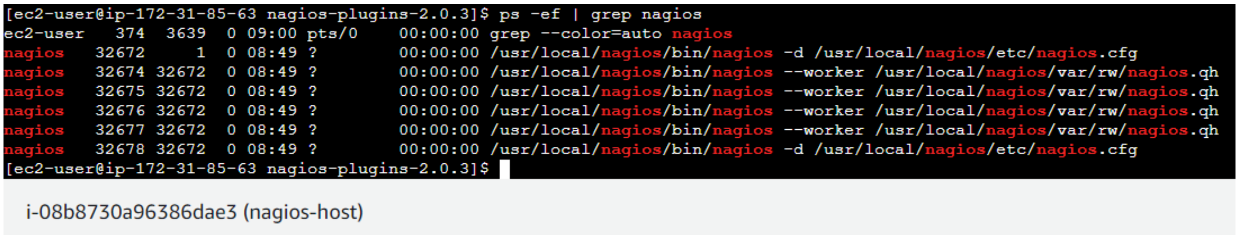


aws
Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia ▼
Mikil Lalwani ▼
Cloud9 IAM EC2 Elastic Beanstalk AWS Amplify Lambda S3
Connect
Instance state ▼
Actions ▼
Launch instances ▼
Find instance by attribute or tag (case-sensitive)
1

Instances (5) Info

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
□	worker2	i-055c803960d6c9721	⊖ Terminated	t2.micro	–	No alarms +	us-east-1a	–
□	worker1	i-08eb10680020c7743	⊖ Terminated	t2.micro	–	No alarms +	us-east-1a	–
□	linux-client	i-0475278b51d26c07c	◷ Pending	t2.micro	–	No alarms +	us-east-1a	ec2-34-228-153-1

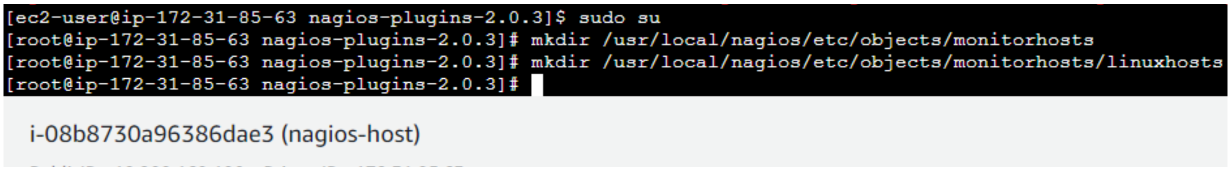
3. On the server, run this command
   ps -ef | grep nagios



```
[ec2-user@ip-172-31-85-63 nagios-plugins-2.0.3]$ ps -ef | grep nagios
ec2-user   374  3639  0 09:00 pts/0    00:00:00 grep --color=auto nagios
nagios   32672     1  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios   32674 32672  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   32675 32672  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   32676 32672  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   32677 32672  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   32678 32672  0 08:49 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
[ec2-user@ip-172-31-85-63 nagios-plugins-2.0.3]$
```

i-08b8730a96386dae3 (nagios-host)

4. Become a root user and create 2 folders
   sudo su
   mkdir /usr/local/nagios/etc/objects/monitorhosts
   mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts

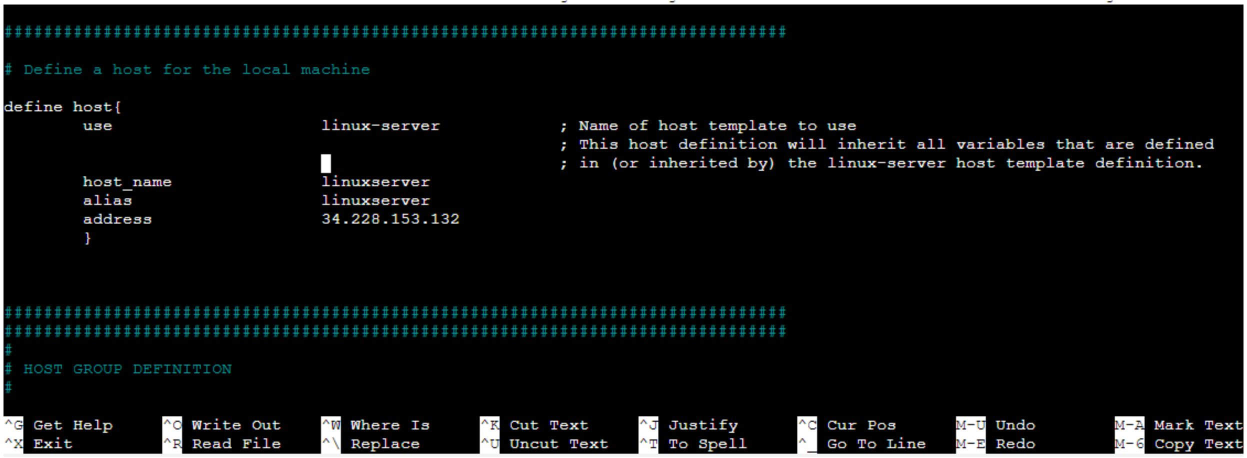


```
[ec2-user@ip-172-31-85-63 nagios-plugins-2.0.3]$ sudo su
[root@ip-172-31-85-63 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-85-63 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-85-63 nagios-plugins-2.0.3]#
```

i-08b8730a96386dae3 (nagios-host)

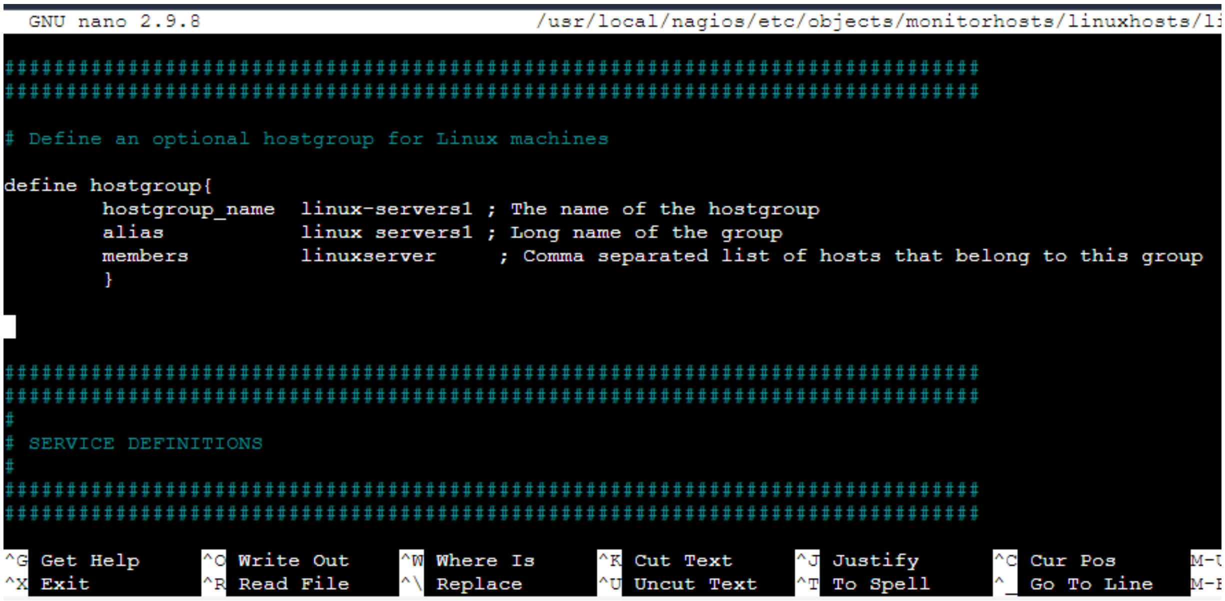
5. Copy the sample localhost.cfg file to linuxhost folder
   cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg

6. Open linuxserver.cfg using nano and make the following changes
   nano /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
   Change the hostname to linuxserver (EVERYWHERE ON THE FILE)
   Change address to the public IP address of your LINUX CLIENT.



```
###############################################################################

# 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                   linuxserver
        address                 34.228.153.132
        }



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

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo       M-A Mark Text
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E Redo       M-6 Copy Text
```

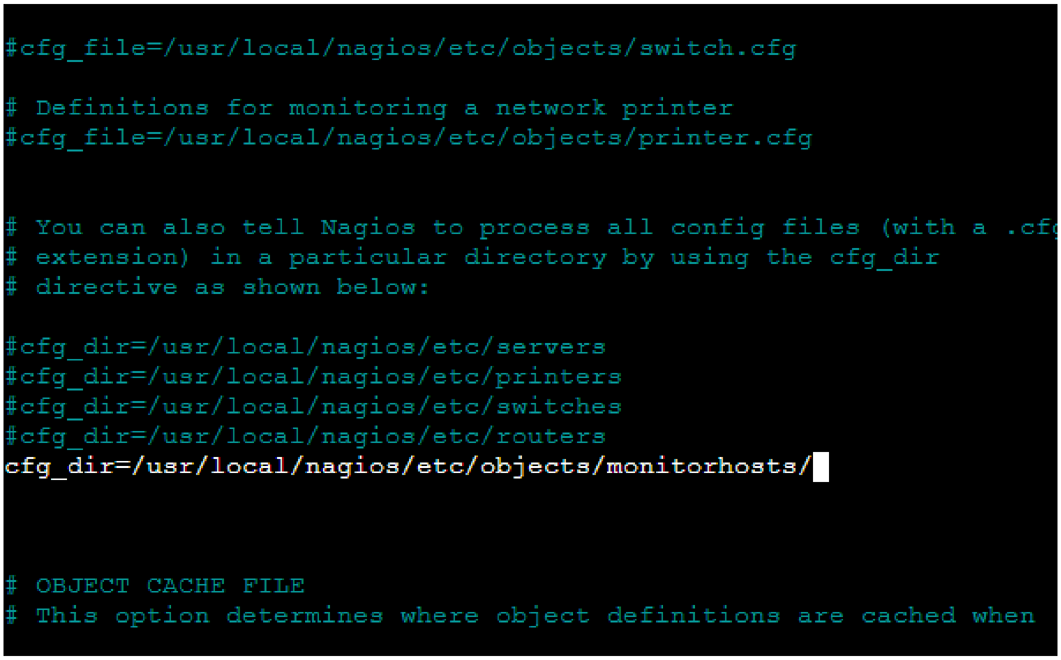
Change hostgroup_name under hostgroup to linux-servers1



```
  GNU nano 2.9.8                     /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/li

###############################################################################
###############################################################################

# Define an optional hostgroup for Linux machines

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



###############################################################################
###############################################################################
#
# SERVICE DEFINITIONS
#
###############################################################################
###############################################################################

^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U
^X Exit        ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line  M-E
```

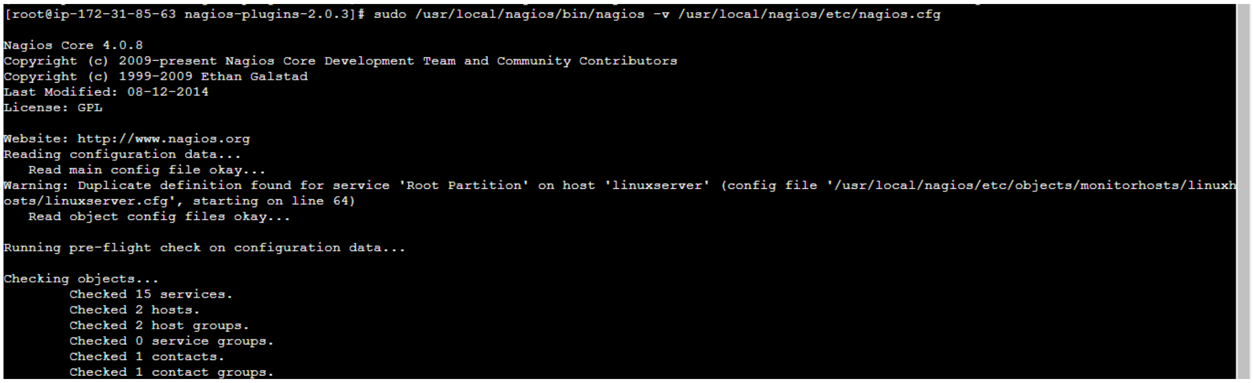
Everywhere else on the file, change the hostname to linuxserver instead of localhost.

7. Open the Nagios Config file and add the following line
   nano /usr/local/nagios/etc/nagios.cfg
   ##Add this line
   cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/



```
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

# Definitions for monitoring a network printer
#cfg_file=/usr/local/nagios/etc/objects/printer.cfg


# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:

#cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/



# OBJECT CACHE FILE
# This option determines where object definitions are cached when
```

8. Verify the configuration files
   sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg



```
[root@ip-172-31-85-63 nagios-plugins-2.0.3]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.0.8
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-12-2014
License: GPL

Website: http://www.nagios.org
Reading configuration data...
   Read main config file okay...
Warning: Duplicate definition found for service 'Root Partition' on host 'linuxserver' (config file '/usr/local/nagios/etc/objects/monitorhosts/linuxh
osts/linuxserver.cfg', starting on line 64)
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 15 services.
        Checked 2 hosts.
        Checked 2 host groups.
        Checked 0 service groups.
        Checked 1 contacts.
        Checked 1 contact groups.
```

   You are good to go if there are no errors.

9. Restart the nagios service

service nagios restart

Now it is time to switch to the client machine.

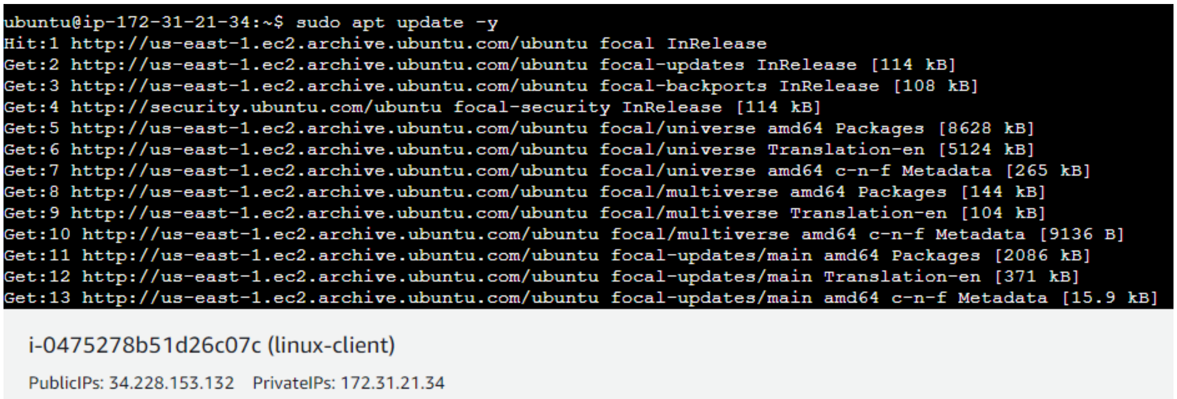
10. SSH into the machine or simply use the EC2 Instance Connect feature.

11. Make a package index update and install gcc, nagios-nrpe-server and the plugins.

    sudo apt update -y

    sudo apt install gcc -y

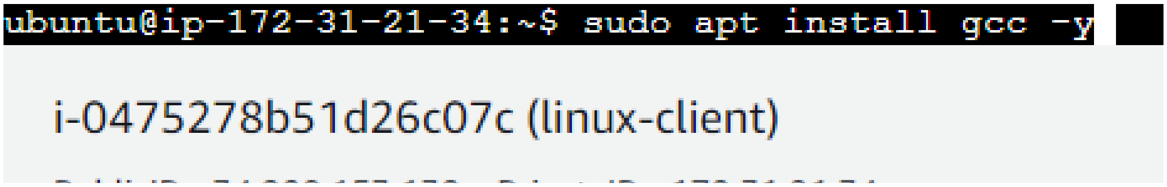
    sudo apt install -y nagios-nrpe-server nagios-plugins



```
ubuntu@ip-172-31-21-34:~$ sudo apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [144 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse Translation-en [104 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 c-n-f Metadata [9136 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [2086 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [371 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [15.9 kB]
```

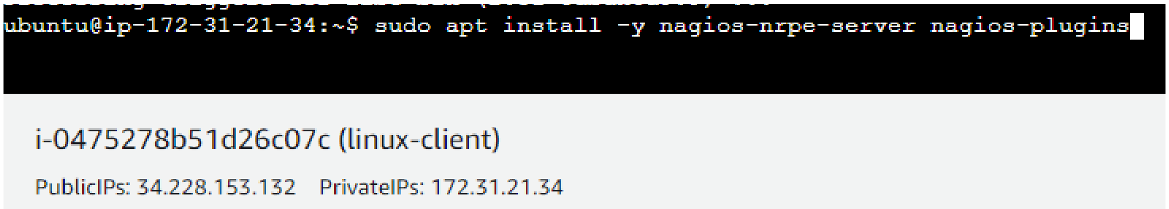
i-0475278b51d26c07c (linux-client)

PublicIPs: 34.228.153.132 PrivateIPs: 172.31.21.34



```
ubuntu@ip-172-31-21-34:~$ sudo apt install gcc -y
```

i-0475278b51d26c07c (linux-client)



```
ubuntu@ip-172-31-21-34:~$ sudo apt install -y nagios-nrpe-server nagios-plugins
```

i-0475278b51d26c07c (linux-client)

PublicIPs: 34.228.153.132 PrivateIPs: 172.31.21.34

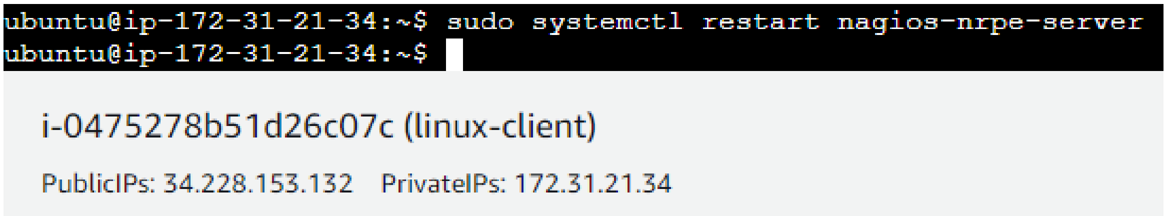
12. Open nrpe.cfg file to make changes.

    sudo nano /etc/nagios/nrpe.cfg

    Under allowed_hosts, add your nagios host IP address like so

13. Restart the NRPE server

    sudo systemctl restart nagios-nrpe-server

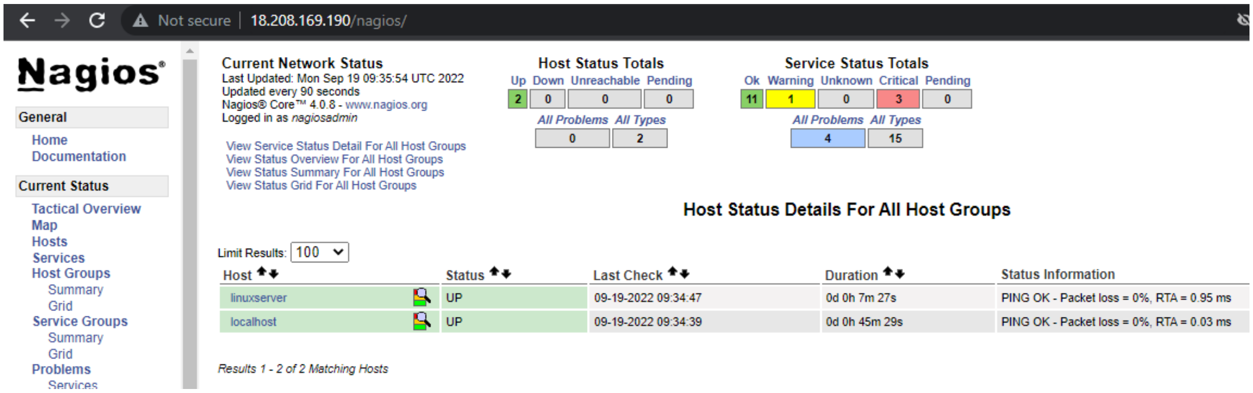


```
ubuntu@ip-172-31-21-34:~$ sudo systemctl restart nagios-nrpe-server
ubuntu@ip-172-31-21-34:~$
```

i-0475278b51d26c07c (linux-client)

PublicIPs: 34.228.153.132 PrivateIPs: 172.31.21.34

14. Now, check your nagios dashboard and you'll see a new host being added. Click on Hosts.



Current Network Status
Last Updated: Mon Sep 19 09:35:54 UTC 2022
Updated every 90 seconds
Nagios® Core™ 4.0.8 - 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

All Problems	All Types
0	2

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
11	1	0	3	0

All Problems	All Types
4	15

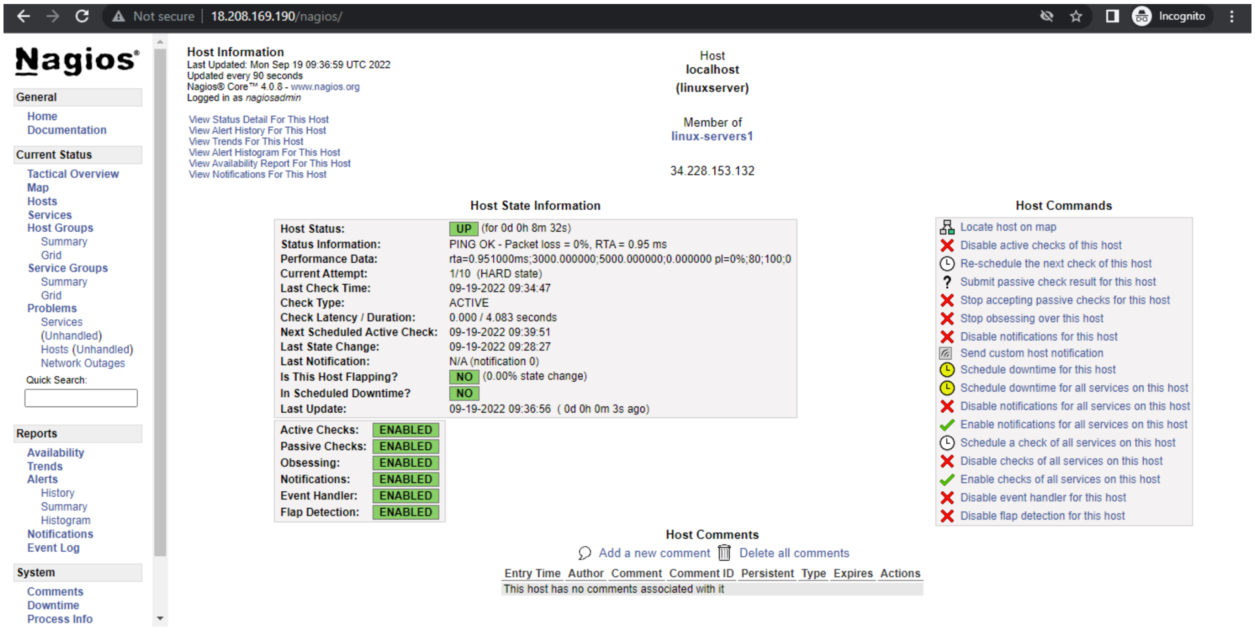
Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-19-2022 09:34:47	0d 0h 7m 27s	PING OK - Packet loss = 0%, RTA = 0.95 ms
localhost	UP	09-19-2022 09:34:39	0d 0h 45m 29s	PING OK - Packet loss = 0%, RTA = 0.03 ms

Results 1 - 2 of 2 Matching Hosts

Click on linuxserver to see the host details



Host Information
Last Updated: Mon Sep 19 09:36:59 UTC 2022
Updated every 90 seconds
Nagios® Core™ 4.0.8 - 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
linux-servers1

34.228.153.132

Host State Information

Host Status:	UP (for 0d 0h 8m 32s)
Status Information:	PING OK - Packet loss = 0%, RTA = 0.95 ms
Performance Data:	rta=0.951000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0
Current Attempt:	1/10 (HARD state)
Last Check Time:	09-19-2022 09:34:47
Check Type:	ACTIVE
Check Latency / Duration:	0.000 / 4.083 seconds
Next Scheduled Active Check:	09-19-2022 09:39:51
Last State Change:	09-19-2022 09:28:27
Last Notification:	N/A (notification 0)
Is This Host Flapping?	NO (0.00% state change)
In Scheduled Downtime?	NO
Last Update:	09-19-2022 09:36:56 (0d 0h 0m 3s ago)

Active Checks:	ENABLED
Passive Checks:	ENABLED
Obsessing:	ENABLED
Notifications:	ENABLED
Event Handler:	ENABLED
Flap Detection:	ENABLED

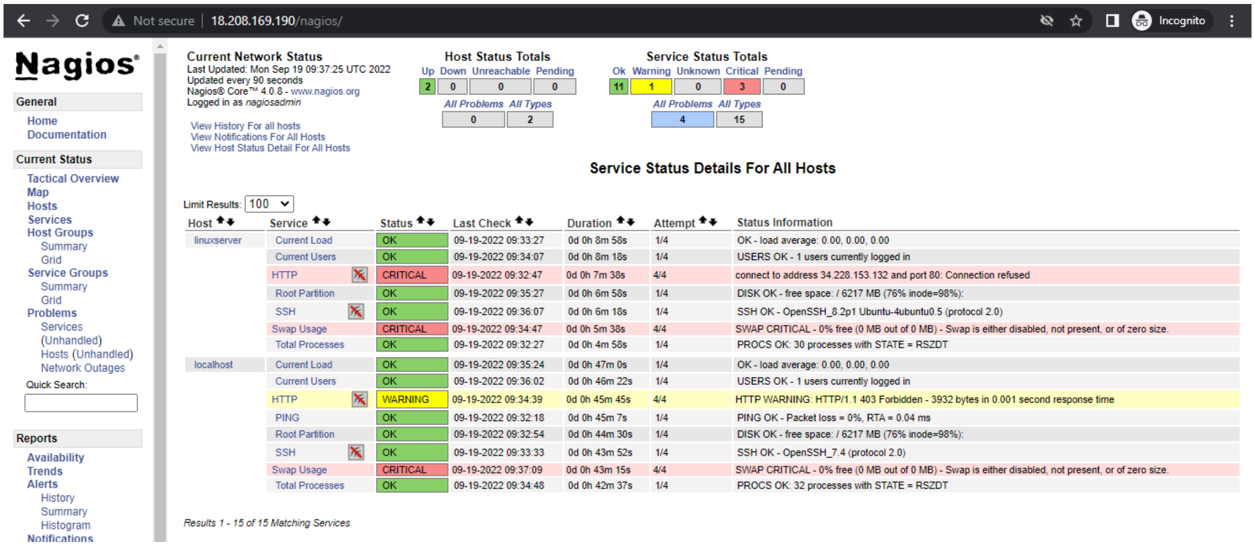
Host Commands

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

Host Comments

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

You can click Services to see all services and ports being monitored.



Current Network Status
Last Updated: Mon Sep 19 09:37:25 UTC 2022
Updated every 90 seconds
Nagios® Core™ 4.0.8 - www.nagios.org
Logged in as *nagiosadmin*

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

All Problems	*All Types*
0	2

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
11	1	0	3	0

All Problems	*All Types*
4	15

Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	09-19-2022 09:33:27	0d 0h 8m 58s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	09-19-2022 09:34:07	0d 0h 8m 18s	1/4	USERS OK - 1 users currently logged in
	HTTP	CRITICAL	09-19-2022 09:32:47	0d 0h 7m 38s	4/4	connect to address 34.228.153.132 and port 80: Connection refused
	Root Partition	OK	09-19-2022 09:35:27	0d 0h 6m 58s	1/4	DISK OK - free space: / 6217 MB (76% inode=98%):
	SSH	OK	09-19-2022 09:36:07	0d 0h 6m 18s	1/4	SSH OK - OpenSSH_8.2p1 Ubuntu-4ubuntu0.5 (protocol 2.0)
	Swap Usage	CRITICAL	09-19-2022 09:34:47	0d 0h 5m 38s	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	09-19-2022 09:32:27	0d 0h 4m 58s	1/4	PROCS OK: 30 processes with STATE = RSZDT
localhost	Current Load	OK	09-19-2022 09:35:24	0d 0h 47m 0s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	09-19-2022 09:36:02	0d 0h 46m 22s	1/4	USERS OK - 1 users currently logged in
	HTTP	WARNING	09-19-2022 09:34:39	0d 0h 45m 45s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 3932 bytes in 0.001 second response time
	PING	OK	09-19-2022 09:32:18	0d 0h 45m 7s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
	Root Partition	OK	09-19-2022 09:32:54	0d 0h 44m 30s	1/4	DISK OK - free space: / 6217 MB (76% inode=98%):
	SSH	OK	09-19-2022 09:33:33	0d 0h 43m 52s	1/4	SSH OK - OpenSSH_7.4 (protocol 2.0)
	Swap Usage	CRITICAL	09-19-2022 09:37:09	0d 0h 43m 15s	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	09-19-2022 09:34:48	0d 0h 42m 37s	1/4	PROCS OK: 32 processes with STATE = RSZDT

Results 1 - 15 of 15 Matching Services

As you can see, we have our linuxserver up and running. It is showing critical status on HTTP due to permission errors and swap because there is no partition created.

In this case, we have monitored -
Servers: 1 linux server
Services: swap
Ports: 22, 80 (ssh, http)
Processes: User status, Current load, total processes, root partition, etc.

Recommended Cleanup

- Terminate both of your EC-2 instances to avoid charges.
- Delete the security group if you created a new one (it won’t affect your bill, you may avoid it).

Conclusion:-

Thus, we learned service monitoring and successfully monitored linux server using Nagios and NRPE.