Prathamesh Palve **D15A-32** **Batch B**

Experiment No 10

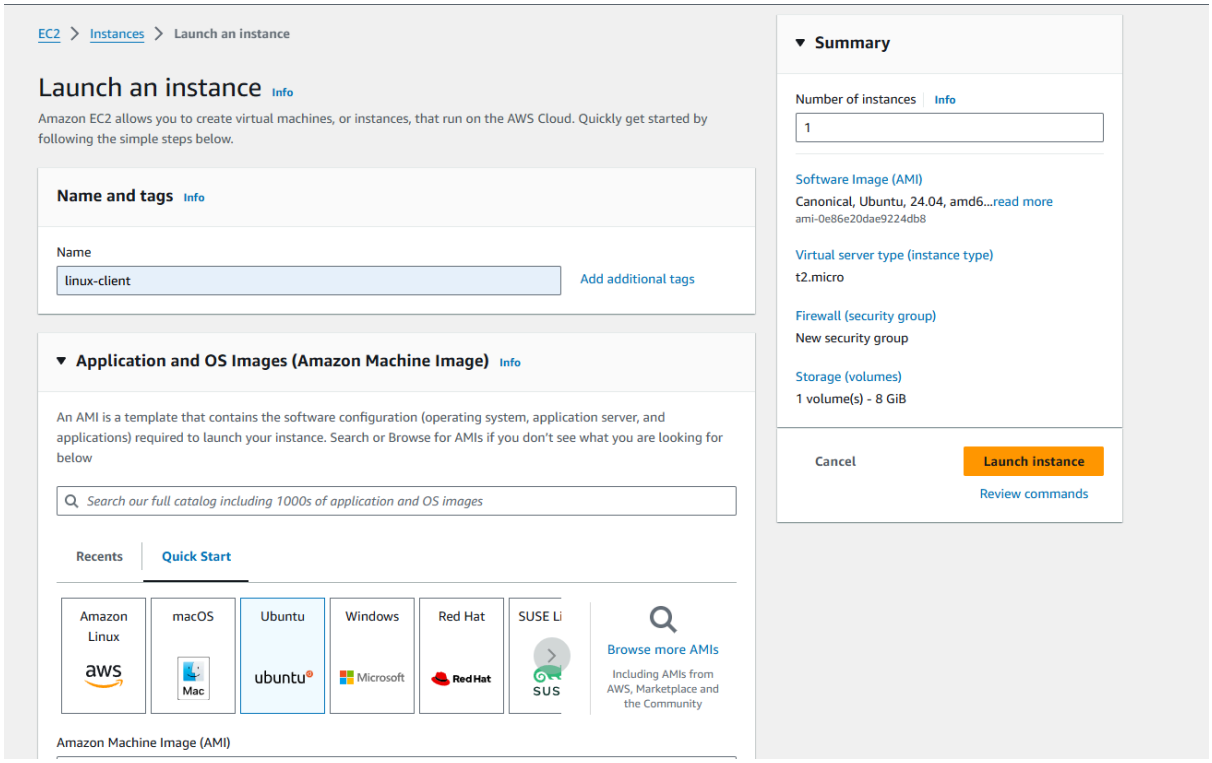
Prerequisites: AWS Free Tier, Nagios Server running on Amazon Linux Machine.

```
PS C:\Users\prath\Desktop\adv DevOps temp> ssh -i "exp-9.pem" ec2-user@3.88.22.82
Last login: Sun Sep 29 20:48:36 2024 from 111.125.237.191
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/
```

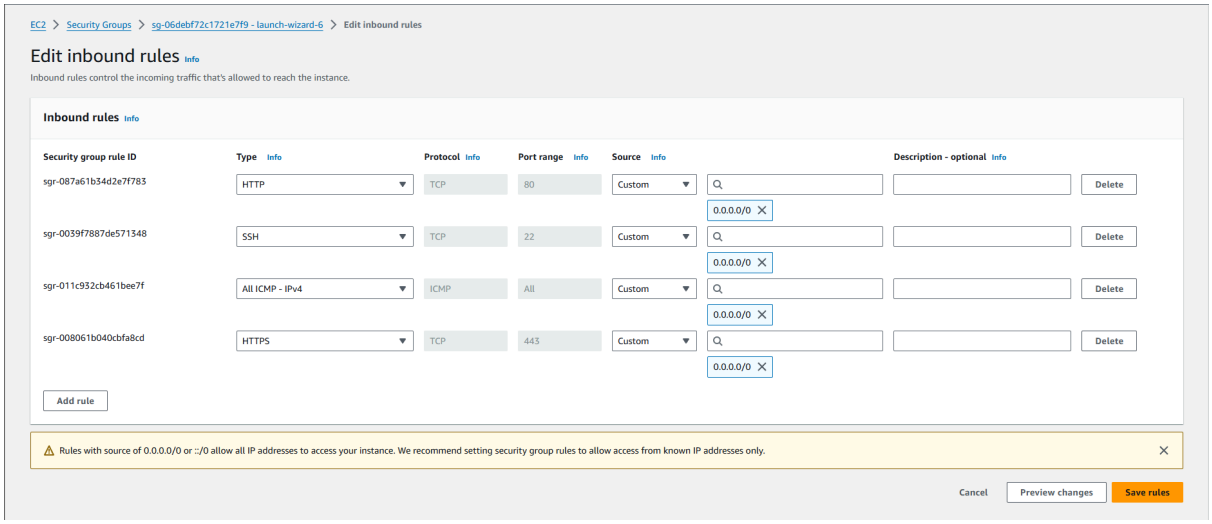
1. To Confirm that Nagios is running on the server side, run this
   sudo systemctl status nagios
   on the “NAGIOS HOST”.

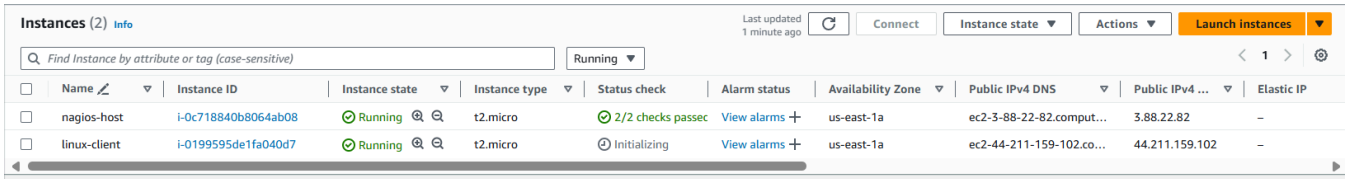
```
[ec2-user@ip-172-31-92-100 ~]$ sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios; bad; vendor preset: disabled)
   Active: active (running) since Sun 2024-09-29 21:06:29 UTC; 1h 5min ago
     Docs: man:systemd-sysv-generator(8)
  Process: 22314 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/nagios.service
           ├─22335 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           ├─22337 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─22338 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─22339 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           ├─22340 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
           └─22341 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
```

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.





3. On the server, run this command

ps -ef | grep nagios

```
[ec2-user@ip-172-31-92-100 ~]$ ps -ef | grep nagios
nagios   22335     1  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios   22337 22335  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   22338 22335  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   22339 22335  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   22340 22335  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios   22341 22335  0 21:06 ?        00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user 31167 31091  0 22:23 pts/1    00:00:00 grep --color=auto nagios
```

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-172-31-92-100 ~]$ sudo su
[root@ip-172-31-92-100 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-92-100 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-92-100 ec2-user]#
```

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

```
[ec2-user@ip-172-31-92-100 ~]$ sudo su
[root@ip-172-31-92-100 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-92-100 ec2-user]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-92-100 ec2-user]# 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                 44.211.159.102
        }
```

Change hostgroup_name under hostgroup to linux-servers1

```
# 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         linuxserver     ; Comma separated list of hosts that belong to this group
        }
```

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/

```
  GNU nano 2.9.8                                  /usr/local/nagios/etc/nagios.

# 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

# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

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

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

```
Checking objects...
        Checked 11 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-172-31-92-100 ec2-user]#
```

9. Restart the nagios service

service nagios restart

```
[root@ip-172-31-92-100 ec2-user]# service nagios restart
Restarting nagios (via systemctl):                         [  OK  ]
[root@ip-172-31-92-100 ec2-user]#
```

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

```
PS C:\Users\prath\Desktop\adv DevOps temp> ssh -i "exp-10.pem" ubuntu@44.211.159.102
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 Sun Sep 29 22:50:12 UTC 2024

  System load:  0.0               Processes:             104
  Usage of /:   22.7% of 6.71GB   Users logged in:       0
  Memory usage: 20%               IPv4 address for enX0: 172.31.93.79
  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-172-31-93-79:~$
```

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

```
Get:43 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe
Get:44 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe
Get:45 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe
Get:46 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe
Get:47 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricte
Get:48 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricte
Get:49 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multivers
Get:50 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multivers
Fetched 29.1 MB in 6s (5068 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
143 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

```
ubuntu@ip-172-31-93-79:~$ sudo apt install gcc nagios-nrpe-server nagios-plugins -y
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:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-13 cpp-13-x86-64-linux-gn
  gcc-13 gcc-13-base gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu libaom3 libasan8 lib
  libc-bin libc-dev-bin libc-devtools libc6 libc6-dev libcc1-0 libcrypt-dev libctf-nob
```

12. Open nrpe.cfg file to make changes.
    sudo nano /etc/nagios/nrpe.cfg

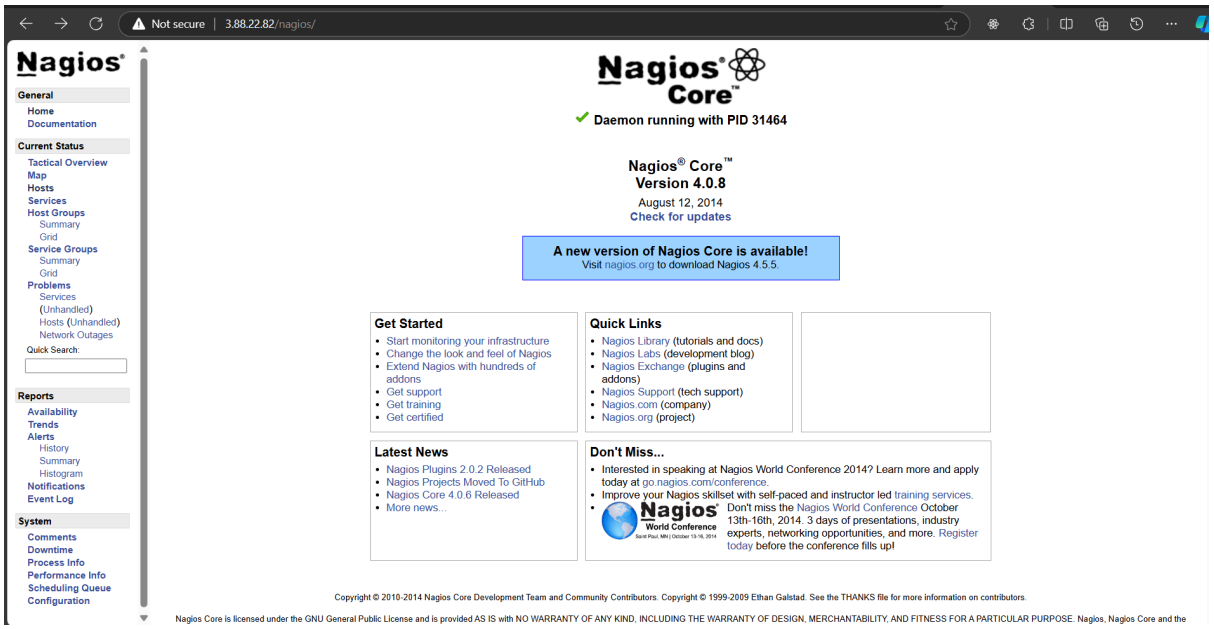
Under allowed_hosts, add your nagios host IP address like so

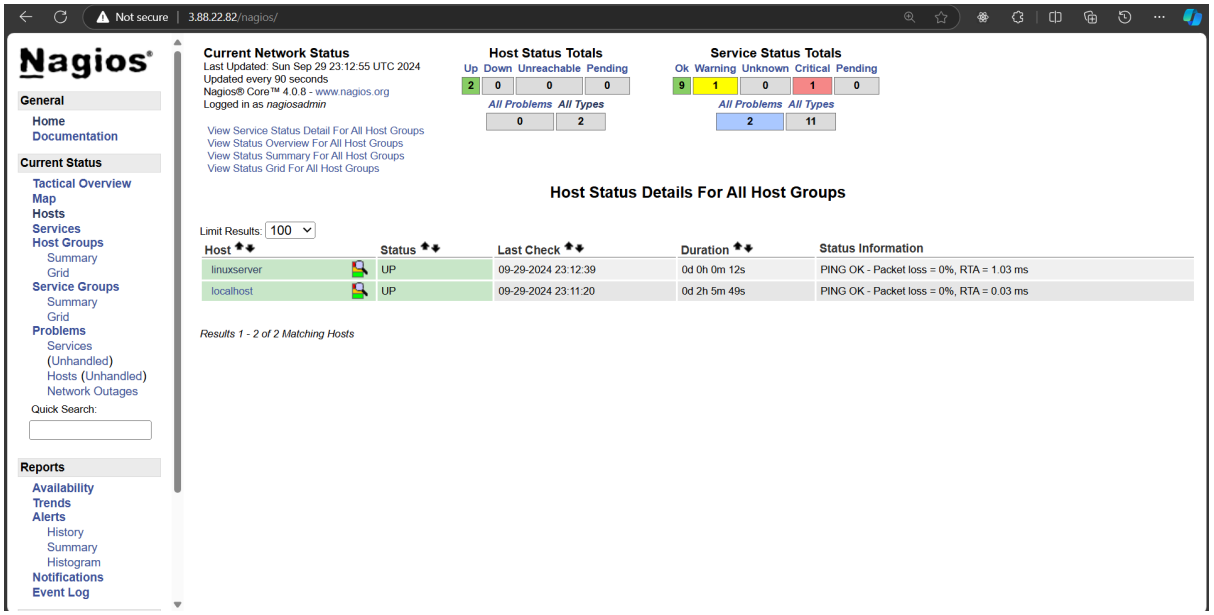
```
  GNU nano 7.2                                        /etc/nagios/nrpe.cfg
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

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=127.0.0.1,3.88.22.82
```

13. Restart the NRPE server
    sudo systemctl restart nagios-nrpe-server

```
ubuntu@ip-172-31-93-79:~$ sudo systemctl restart nagios-nrpe-server
```

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





15. Click on linuxserver to see the host details