Name : Nidhi Rijhwani
Subject : Adv Devops Exp No 09

Aim : To Understand Continuous monitoring and Installation and configuration of Nagios Core, Nagios Plugins and NRPE (Nagios Remote Plugin Executor) on Linux Machine.

Theory:

What is Nagios?

Nagios is an open-source software for continuous monitoring of systems, networks, and infrastructures. It runs plugins stored on a server that is connected with a host or another server on your network or the Internet. In case of any failure, Nagios alerts about the issues so that the technical team can perform the recovery process immediately. Nagios is used for continuous monitoring of systems, applications, service and business processes in a DevOps culture.

Why Do We Need Nagios Tools?

Here are the important reasons to use Nagios monitoring tool:

- Detects all types of network or server issues.
- Helps you to find the root cause of the problem which allows you to get the permanent solution to the problem.
- Active monitoring of your entire infrastructure and business processes.
- Allows you to monitor and troubleshoot server performance issues.
- Helps you to plan for infrastructure upgrades before outdated systems create failures.
- You can maintain the security and availability of the service.
- Automatically fix problems in a panic situation.

Features of Nagios

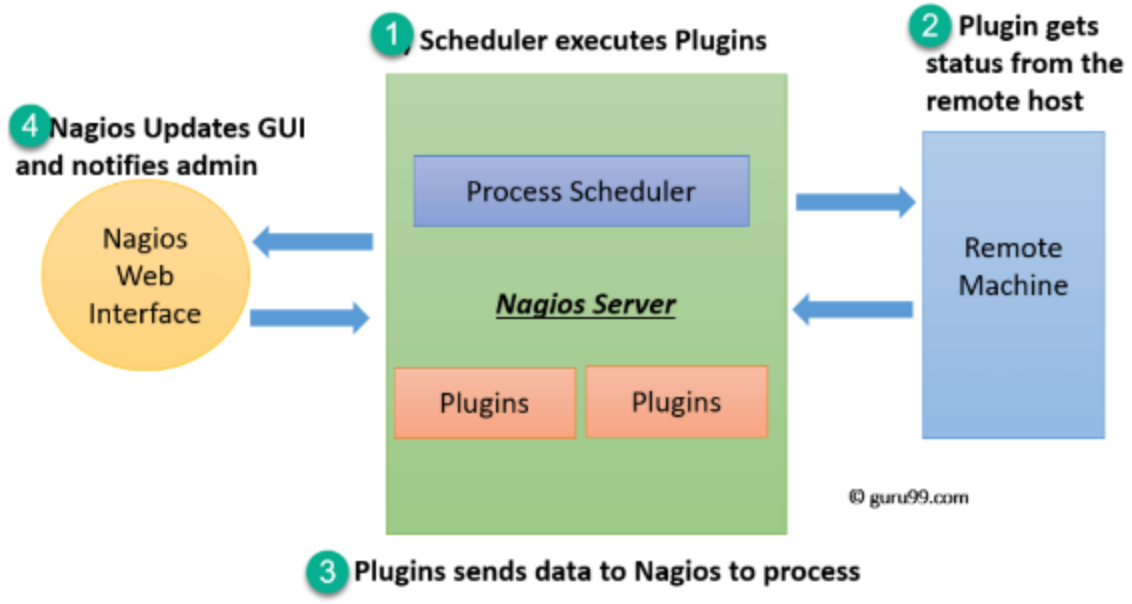
Following are the important features of Nagios monitoring tool:

- Relatively scalable, Manageable, and Secure.
- Good log and database system.
- Informative and attractive web interfaces.
- Automatically send alerts if condition changes.
- If the services are running fine, then there is no need to check that the host is alive.
- Helps you to detect network errors or server crashes.
- You can troubleshoot the performance issues of the server.
- The issues, if any, can be fixed automatically as they are identified during the monitoring process.

- You can monitor the entire business process and IT infrastructure with a single pass.
- The product's architecture is easy to write new plugins in the language of your choice.
- Nagios allows you to read its configuration from an entire directory which helps you to decide how to define individual files.
- Utilizes topology to determine dependencies.
- Monitor network services like HTTP, SMTP, HTTP, SNMP, FTP, SSH, POP, etc.
- Helps you to define network host hierarchy using parent hosts.
- Ability to define event handlers that run during service or host events for proactive problem resolution.
- Support for implementing redundant monitoring hosts.

Nagios Architecture

Nagios is a client-server architecture. Usually, on a network, a Nagios server is running on a host, and plugins are running on all the remote hosts which should be monitored.



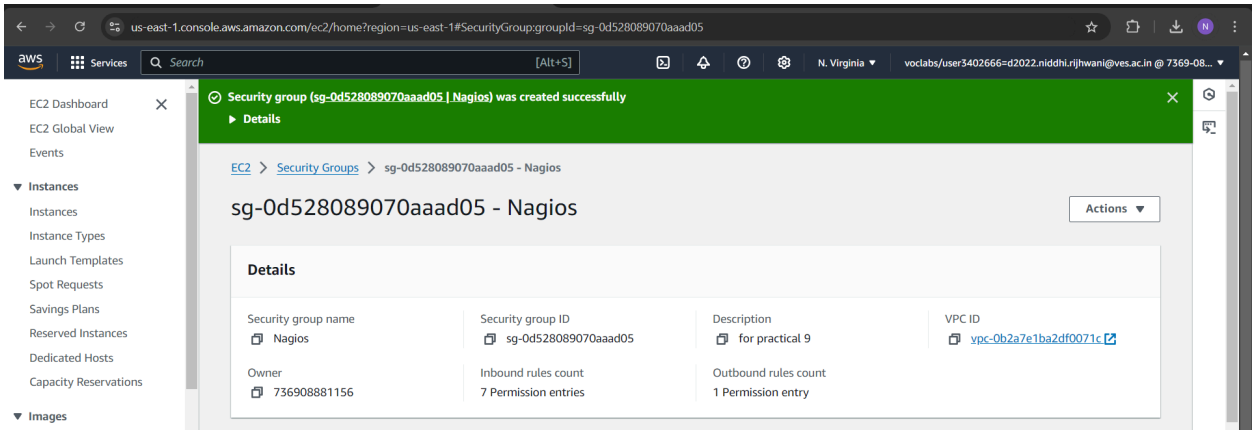
1. The scheduler is a component of the server part of Nagios. It sends a signal to execute the plugins at the remote host.
2. The plugin gets the status from the remote host.
3. The plugin sends the data to the process scheduler.
4. The process scheduler updates the GUI and notifications are sent to admins.

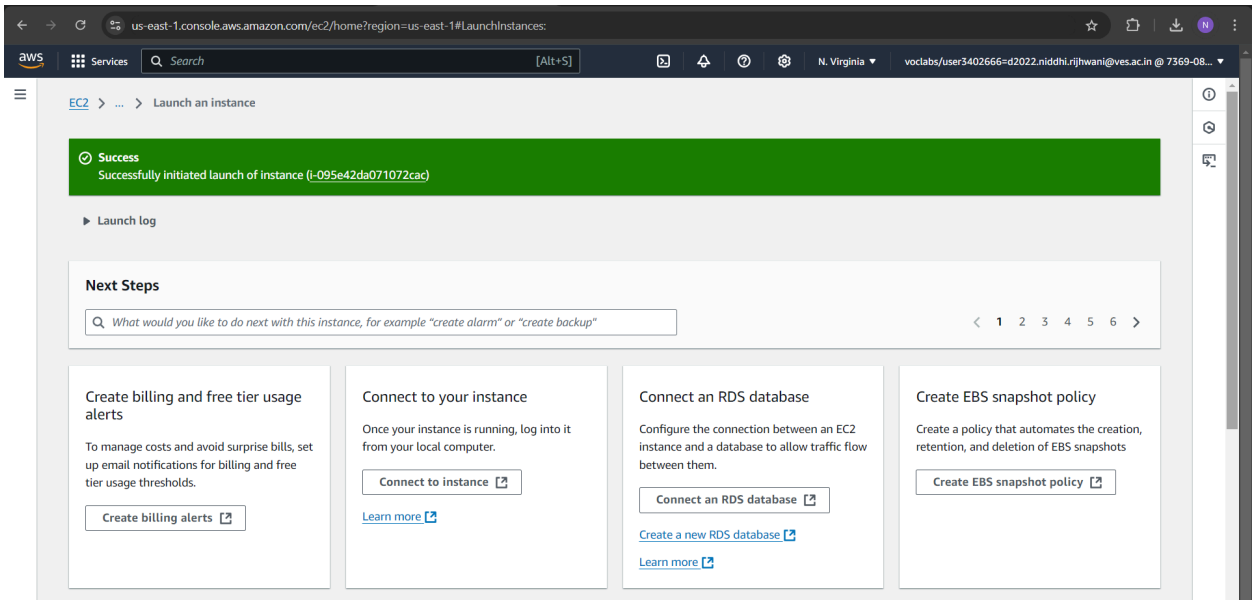
Installation of Nagios

Prerequisites: AWS Free Tier

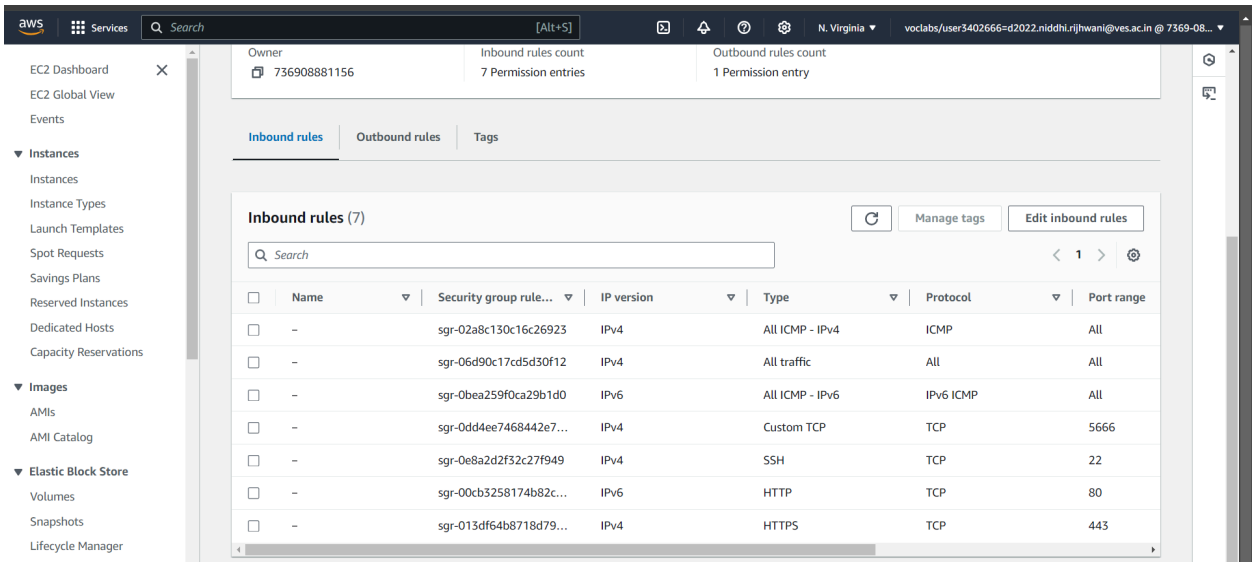
Steps:

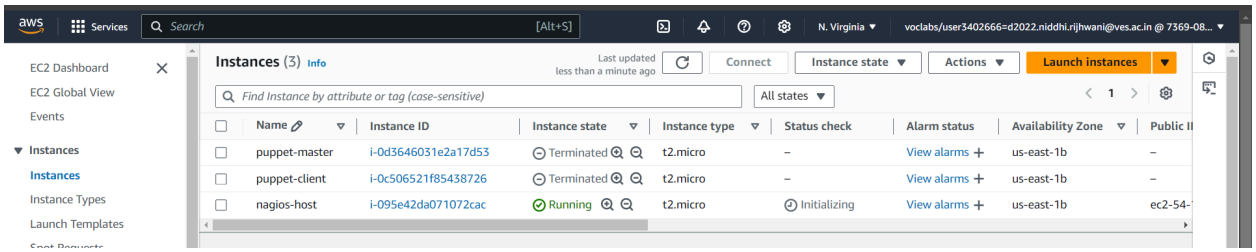
1. Create an Amazon Linux EC2 Instance in AWS and name it - nagios-host.



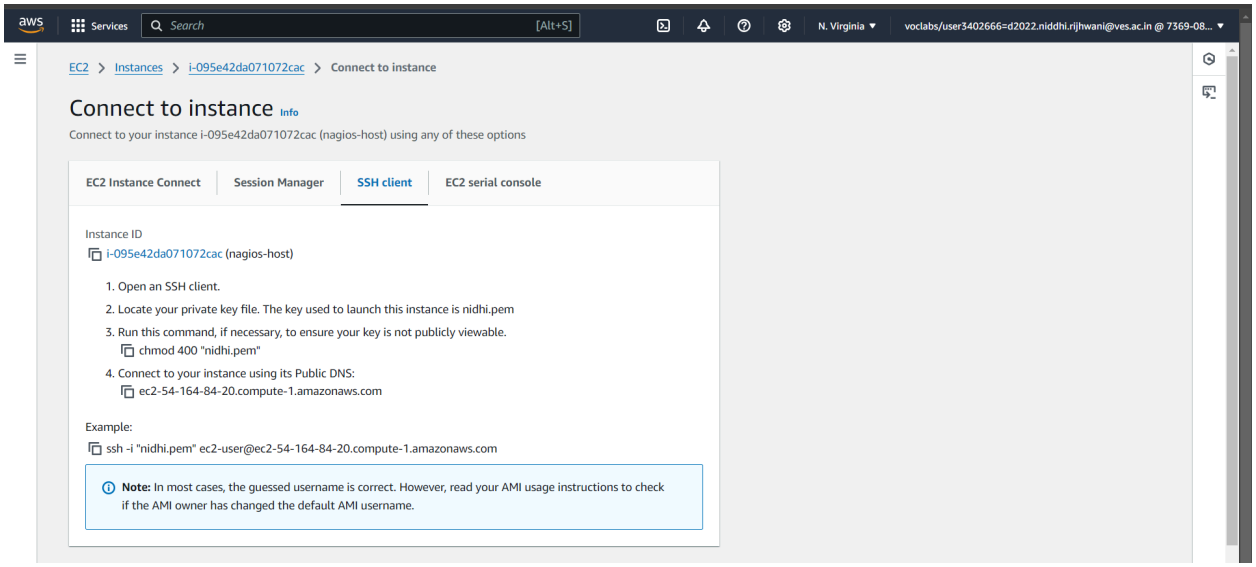


2. Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere. We have to edit the inbound of the specific security group.





3. SSH into your EC2 instance or simply use EC2 instance Connect from the browser.



4. Update the package indices and install the following packages using yum

- sudo yum update
- sudo yum install httpd php
- sudo yum install gcc glibc glibc-common
- sudo yum install gd gd-devel

```
PS C:\Users\rijhw\Downloads> ssh -i "nidhi.pem" ec2-user@ec2-54-164-84-20.compute-1.amazonaws.com
The authenticity of host 'ec2-54-164-84-20.compute-1.amazonaws.com (54.164.84.20)' can't be established.
ED25519 key fingerprint is SHA256:yCk/5jk8rtieC1k87PEXPimwNU9kLmm6cZrwEHaxD30.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-164-84-20.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-31-40-81 ~]$ sudo yum install httpd php
sudo yum install gcc glibc glibc-common
sudo yum install gd gd-devel
Last metadata expiration check: 0:03:20 ago on Sun Oct 13 07:45:06 2024.
Dependencies resolved.
================================================================================
 Package              Arch    Version                     Repository    Size
================================================================================
Installing:
 httpd                x86_64  2.4.62-1.amzn2023           amazonlinux   48 k
 php8.3               x86_64  8.3.10-1.amzn2023.0.1       amazonlinux   10 k
Installing dependencies:
 apr                  x86_64  1.7.2-2.amzn2023.0.2        amazonlinux  129 k
 apr-util             x86_64  1.6.3-1.amzn2023.0.1        amazonlinux   98 k
```

```
  xz-devel-5.2.5-9.amzn2023.0.2.x86_64
  zlib-devel-1.2.11-33.amzn2023.0.5.x86_64

Complete!
[ec2-user@ip-172-31-40-81 ~]$
```

5. Create a new Nagios User with its password. You'll have to enter the password twice for confirmation.
   - sudo adduser -m nagios
   - sudo passwd nagios
6. Create a new user group
   - sudo groupadd nagcmd
7. Use these commands so that you don't have to use sudo for Apache and Nagios
   - sudo usermod -a -G nagcmd nagios
   - sudo usermod -a -G nagcmd apache
8. Create a new directory for Nagios downloads
   - mkdir ~/downloads
   - cd ~/downloads

```
[ec2-user@ip-172-31-40-81 ~]$ sudo adduser -m nagios
sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-40-81 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-40-81 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-40-81 ~]$ mkdir ~/downloads
cd ~/downloads
```

9. Use wget to download the source zip files.

wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.tar.gz

```
[ec2-user@ip-172-31-40-81 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.tar.gz
--2024-10-13 07:53:38--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fef7:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11339450 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.9.tar.gz'

nagios-4.4.9.tar.gz         100%[=======================================>]  10.81M  13.6MB/s
   in 0.8s

2024-10-13 07:53:39 (13.6 MB/s) - 'nagios-4.4.9.tar.gz' saved [11339450/11339450]

[ec2-user@ip-172-31-40-81 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2024-10-13 07:54:27--  http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2659772 (2.5M) [application/x-gzip]
Saving to: 'nagios-plugins-2.0.3.tar.gz'

nagios-plugins-2.0.3.ta 100%[=============================>]   2.54M  6.63MB/s    in 0.4s

2024-10-13 07:54:28 (6.63 MB/s) - 'nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]
```

10. Use tar to unzip and change to that directory.
    - tar zxvf nagios-4.0.8.tar.gz

```
[ec2-user@ip-172-31-40-81 downloads]$ tar zxvf nagios-4.4.9.tar.gz
nagios-4.4.9/
nagios-4.4.9/.gitignore
```

11. Run the configuration script with the same group name you previously created.
    - ./configure --with-command-group=nagcmd

```
[ec2-user@ip-172-31-40-81 downloads]$ cd nagios-4.4.9/
[ec2-user@ip-172-31-40-81 nagios-4.4.9]$ ./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
```

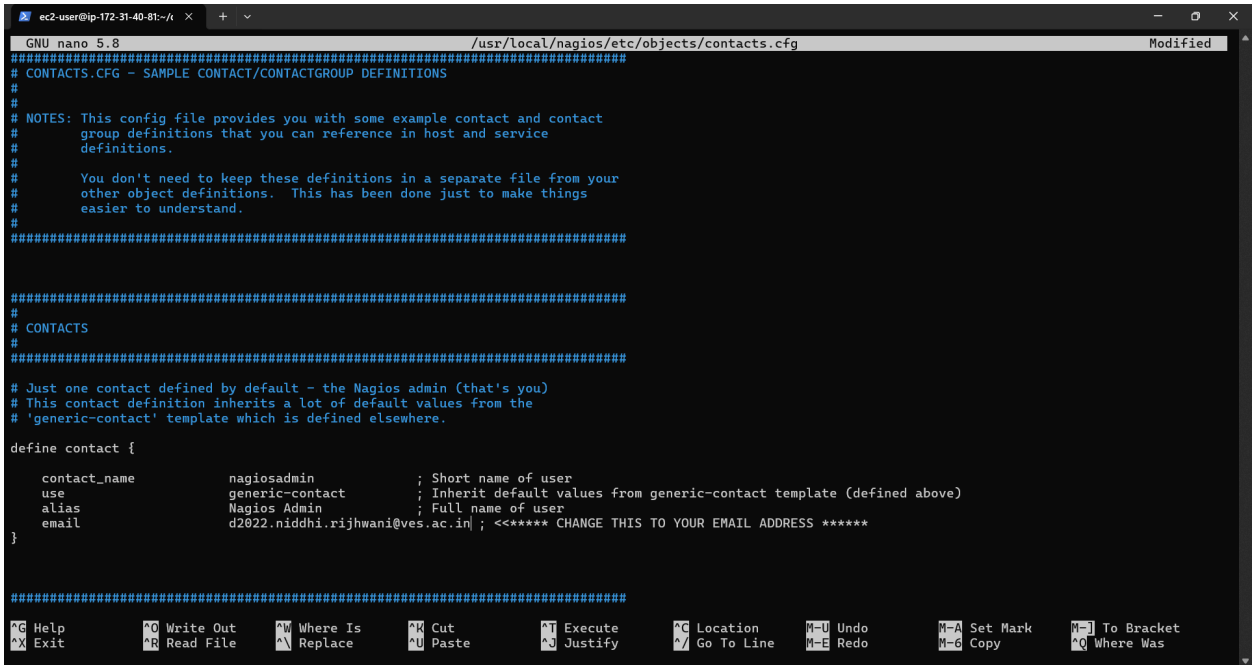
12. Compile the source code.
    - make all

```
[ec2-user@ip-172-31-40-81 nagios-4.4.9]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/base'
gcc -Wall -I.. -g -O2  -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
gcc -Wall -I.. -g -O2  -DHAVE_CONFIG_H -DNSCORE   -c -o broker.o broker.c
gcc -Wall -I.. -g -O2  -DHAVE_CONFIG_H -DNSCORE   -c -o nebmods.o nebmods.c
```

13. Install binaries, init script and sample config files. Lastly, set permissions on the external command directory.
    - sudo make install
    - sudo make install-init
    - sudo make install-config
    - sudo make install-commandmode

```
[ec2-user@ip-172-31-40-81 nagios-4.4.9]$ sudo make install
sudo make install-init
sudo make install-config
sudo make install-commandmode
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -s -m 774 -o nagios -g nagios nagiostats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
```

14. Edit the config file and change the email address.
    - sudo nano /usr/local/nagios/etc/objects/contacts.cfg

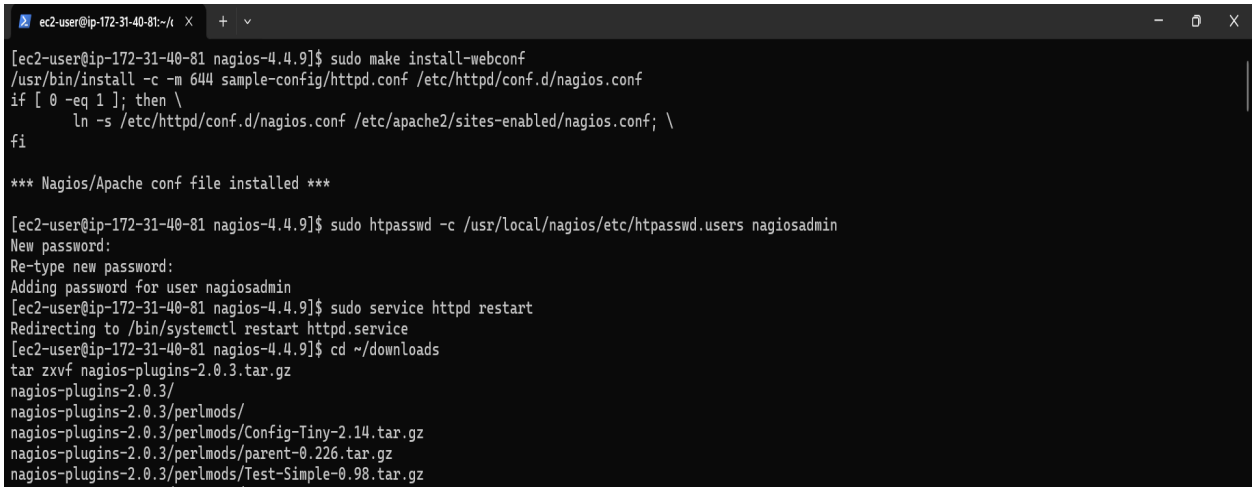


15. Configure the web interface.
    - sudo make install-webconf

16. Create a nagiosadmin account for nagios login along with password. You'll have to specify the password twice.
    - sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin

17. Restart Apache
    - sudo service httpd restart

18. Go back to the downloads folder and unzip the plugins zip file.
    - cd ~/downloads
    - tar zxvf nagios-plugins-2.0.3.tar.gz



19. Compile and install plugins

- cd nagios-plugins-2.0.3
- ./configure --with-nagios-user=nagios --with-nagios-group=nagios make
- sudo make install

```
[ec2-user@ip-172-31-40-81 downloads]$ cd nagios-plugins-2.0.3
./configure --with-nagios-user=nagios --with-nagios-group=nagios
make
sudo make install
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether to disable maintainer-specific portions of Makefiles... yes
checking build system type... x86_64-unknown-linux-gnu
```

20. Start Nagios

Add Nagios to the list of system services

- sudo chkconfig --add nagios
- sudo chkconfig nagios on

Verify the sample configuration files

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

If there are no errors, you can go ahead and start Nagios.

- sudo service nagios start

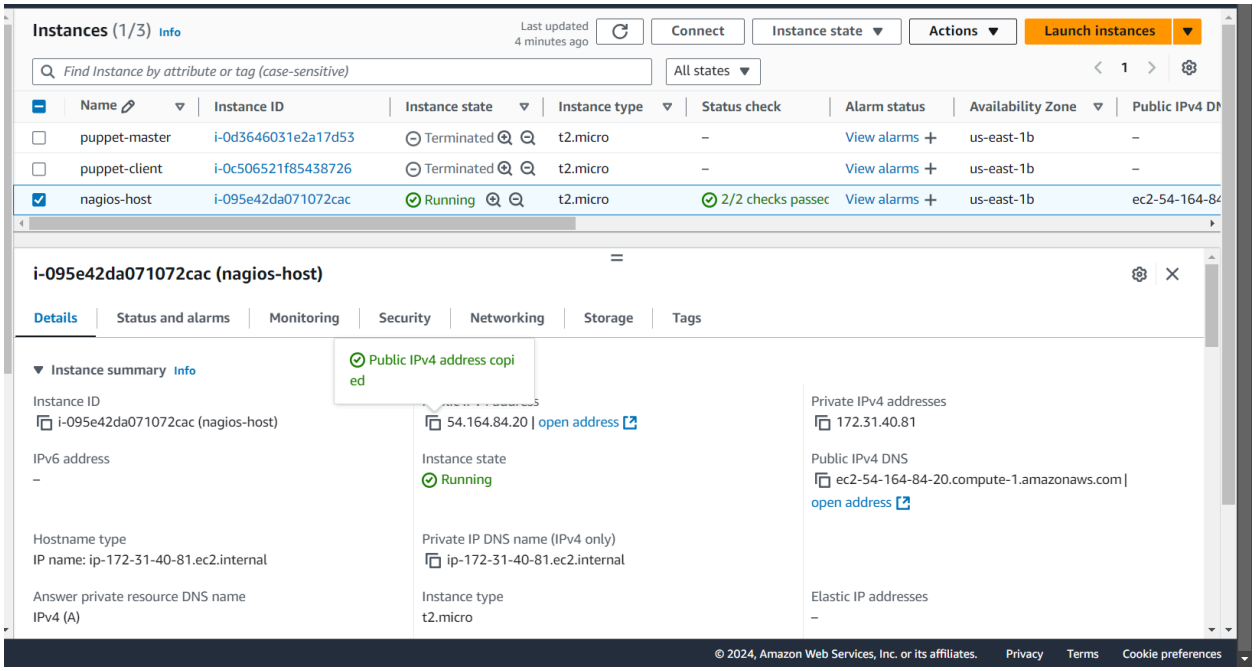
21. Check the status of Nagios

- sudo systemctl status nagios

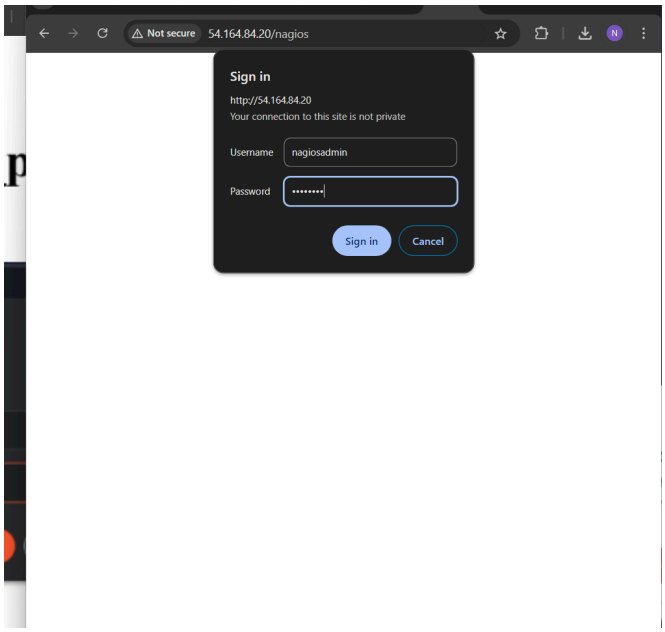
```
[ec2-user@ip-172-31-40-81 nagios-plugins-2.0.3]$ sudo chkconfig --add nagios
error reading information on service nagios: No such file or directory
[ec2-user@ip-172-31-40-81 nagios-plugins-2.0.3]$ sudo chkconfig nagios on
Note: Forwarding request to 'systemctl enable nagios.service'.
[ec2-user@ip-172-31-40-81 nagios-plugins-2.0.3]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-40-81 nagios-plugins-2.0.3]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.9
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Sun 2024-10-13 08:19:54 UTC; 34s ago
       Docs: https://www.nagios.org/documentation
    Process: 104582 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/>
    Process: 104583 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nag>
   Main PID: 104584 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.8M
        CPU: 80ms
     CGroup: /system.slice/nagios.service
             ├─104584 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─104585 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na>
             ├─104586 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na>
             ├─104587 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na>
             ├─104588 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/na>
             └─104589 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 13 08:19:54 ip-172-31-40-81.ec2.internal nagios[104584]: qh: Socket '/usr/local/nag>
```

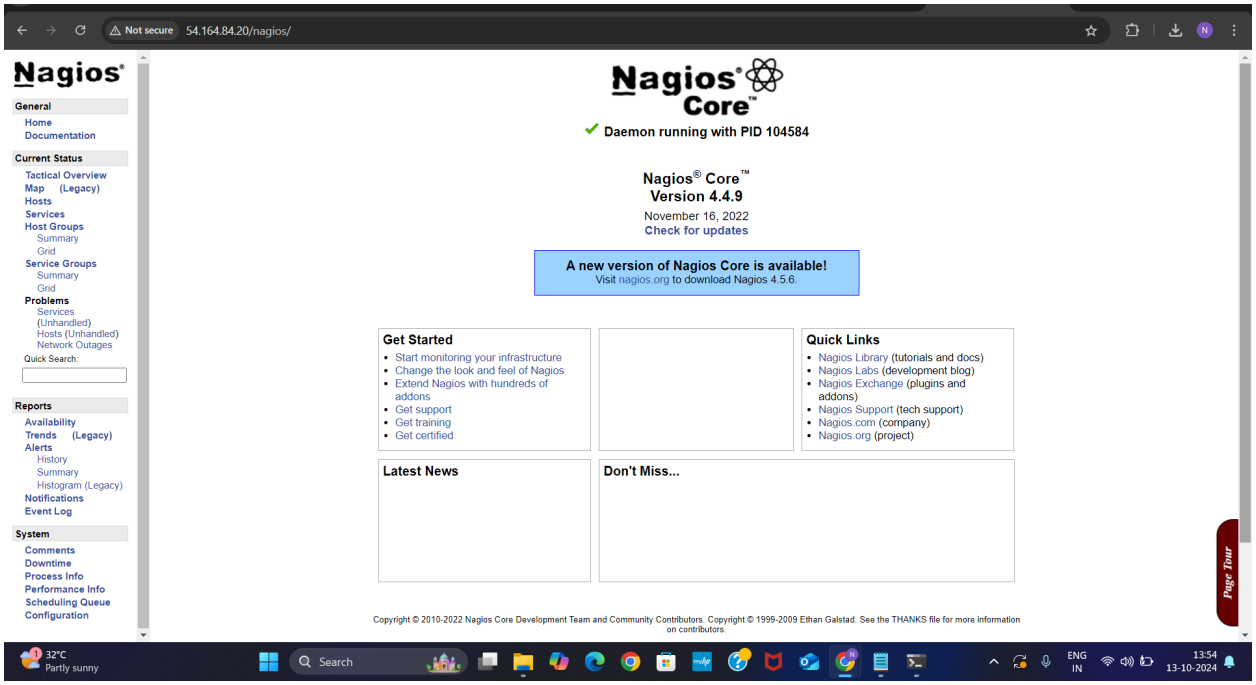
22. Go back to EC2 Console and copy the Public IP address of this instance.



23. Open up your browser and look for http://<your_public_ip_address>/nagios
Enter username as nagiosadmin and password which you set in Step 16.



24. After entering the correct credentials, you will see this page.



This means that Nagios was correctly installed and configured with its plugins so far.

Conclusion:

Thus, we learned about Nagios and successfully set it up as a host on our Amazon Linux machine.