Name:Aarya Gandhi

Roll no.15

Class D15B

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

Theory:

**Nagios Monitoring Tool**

**What is Nagios?**

Nagios is an open-source software for continuous monitoring of systems, networks, and infrastructures. It operates by running plugins stored on a server that connects with a host or another server on your network or the Internet. In case of any failure, Nagios alerts the technical team about the issues, enabling them to perform the recovery process immediately. Nagios is widely used for continuous monitoring of systems, applications, services, and business processes in a DevOps culture.

**Why We Need Nagios**

Here are the important reasons to use the Nagios monitoring tool:

* **Detection of Issues**: Detects all types of network or server issues.
* **Root Cause Analysis**: Helps identify the root cause of problems, allowing for permanent solutions.
* **Active Monitoring**: Provides active monitoring of your entire infrastructure and business processes.
* **Performance Troubleshooting**: Allows you to monitor and troubleshoot server performance issues.
* **Infrastructure Planning**: Helps plan for infrastructure upgrades before outdated systems create failures.
* **Security and Availability**: Maintains the security and availability of services.
* **Automatic Problem Resolution**: Can automatically fix problems in panic situations.

**Features of Nagios**

Following are the important features of the Nagios monitoring tool:

* **Scalability and Security**: Relatively scalable, manageable, and secure.
* **Logging and Database System**: Provides a good log and database system.
* **Web Interface**: Features informative and attractive web interfaces.
* **Automated Alerts**: Automatically sends alerts if conditions change.
* **Host Check Optimization**: If services are running fine, there’s no need to check if the host is alive.
* **Error Detection**: Helps detect network errors or server crashes.
* **Performance Troubleshooting**: Facilitates troubleshooting of server performance issues.
* **Automated Issue Resolution**: Identifies and resolves issues during the monitoring process automatically.
* **Comprehensive Monitoring**: Monitors the entire business process and IT infrastructure with a single pass.
* **Plugin Development**: Easy to write new plugins in the language of your choice.
* **Configuration Management**: Allows reading configuration from an entire directory for easier management of individual files.
* **Topology Utilization**: Utilizes topology to determine dependencies.
* **Service Monitoring**: Monitors network services like HTTP, SMTP, SNMP, FTP, SSH, POP, etc.
* **Network Host Hierarchy**: Helps define network host hierarchy using parent hosts.
* **Event Handlers**: Ability to define event handlers that run during service or host events for proactive problem resolution.
* **Redundant Monitoring Support**: Supports implementing redundant monitoring hosts.

**Nagios Architecture**

Nagios operates on a client-server architecture. Typically, a Nagios server runs on a host, and plugins operate on all the remote hosts that need monitoring.

1. **Scheduler**: The scheduler, a component of the server part of Nagios, sends a signal to execute the plugins on the remote host.
2. **Status Retrieval**: The plugin retrieves the status from the remote host.
3. **Data Transmission**: The plugin sends the data to the process scheduler.
4. **GUI Update and Notifications**: The process scheduler updates the GUI, and notifications are sent to administrators.

Steps:

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



1. Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere.
2. SSH into Your EC2 instance or simply use EC2 Instance Connect from the browser. 
3. 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

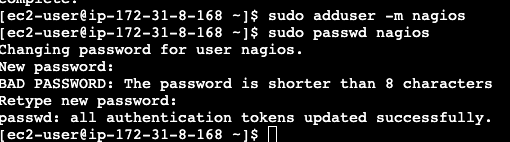
sudo yum install gd gd-devel

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



1. 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

Create a new directory for Nagios downloads

mkdir ~/downloads

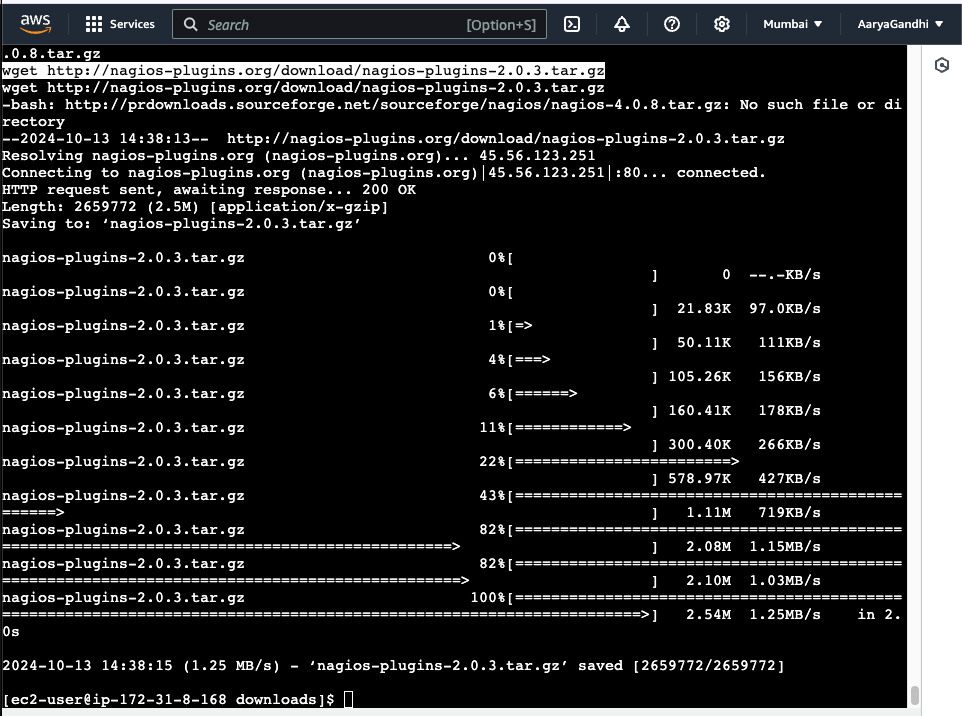
cd ~/downloads

9. Use wget to download the source zip files.

wget

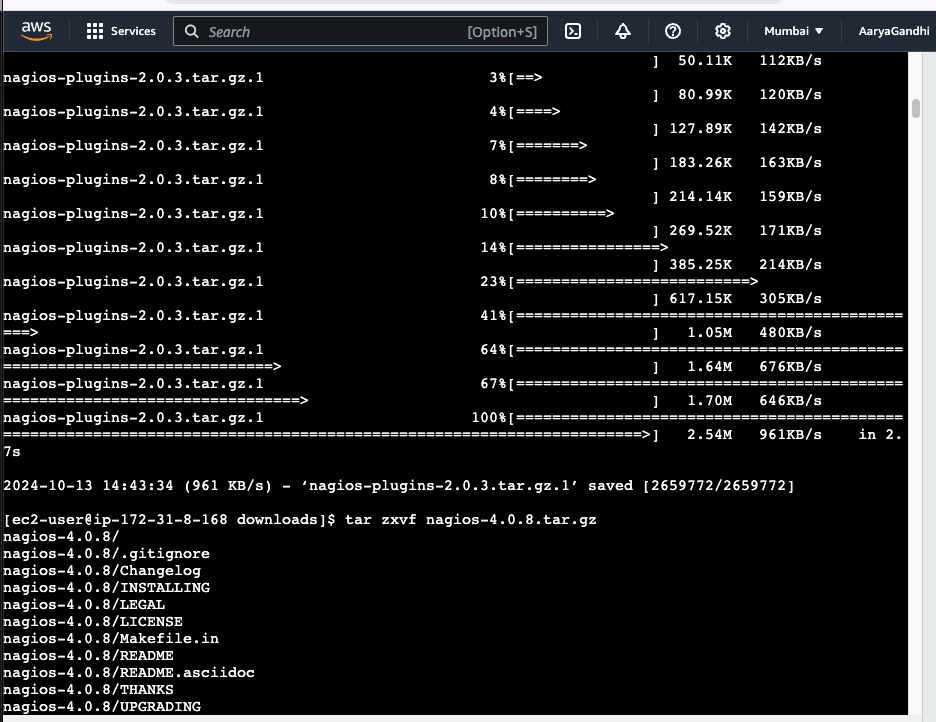
<http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz>

wget <http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz>



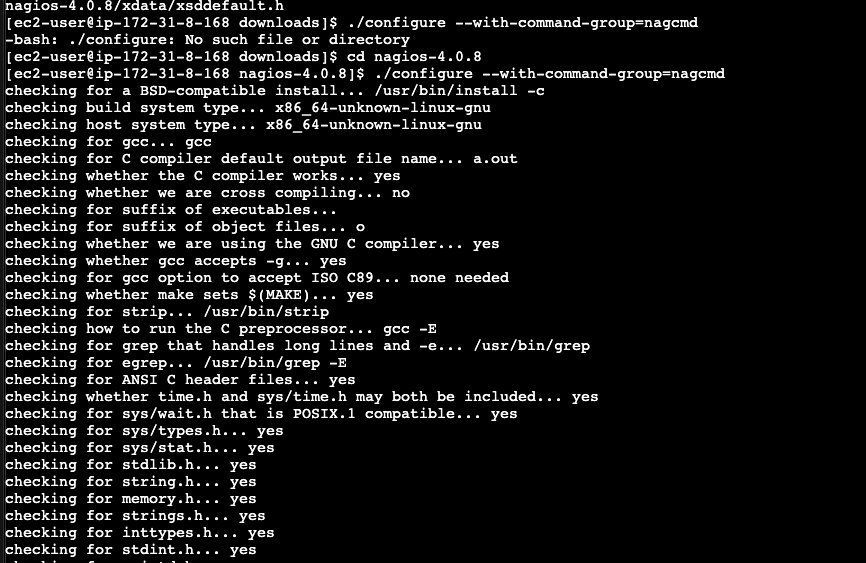
10. Use tar to unzip and change to that directory.

tar zxvf nagios-4.0.8.tar.gz

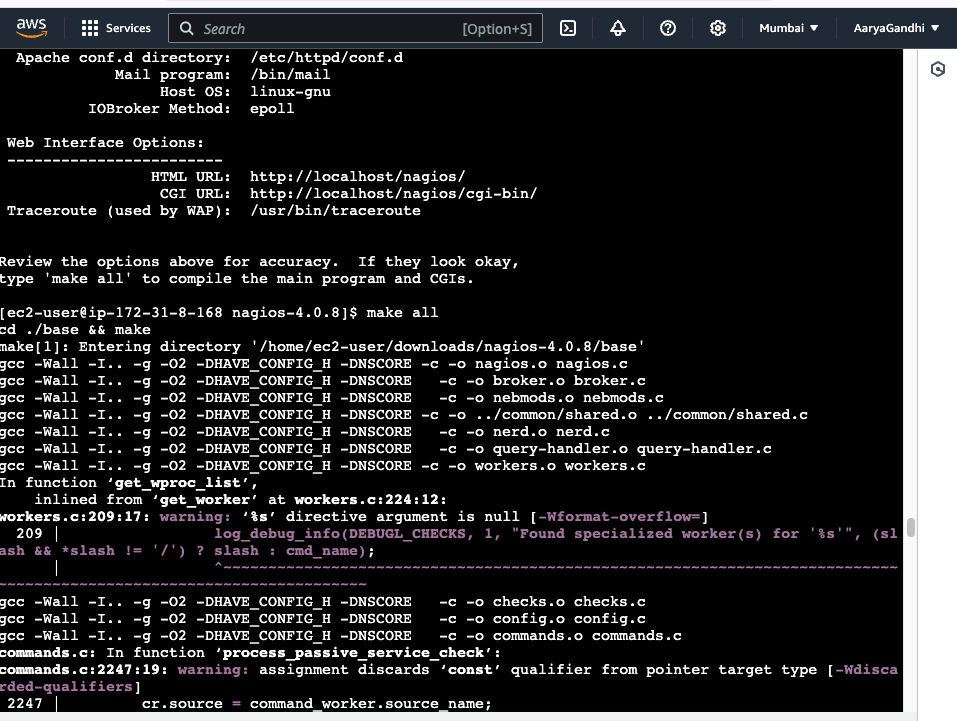


11. Run the configuration script with the same group name you previously created.

./configure --with-command-group=nagcmd



12. Compile the source code.



13. Install binaries, init script and sample config files. Lastly, set permissions on the external

command directory.

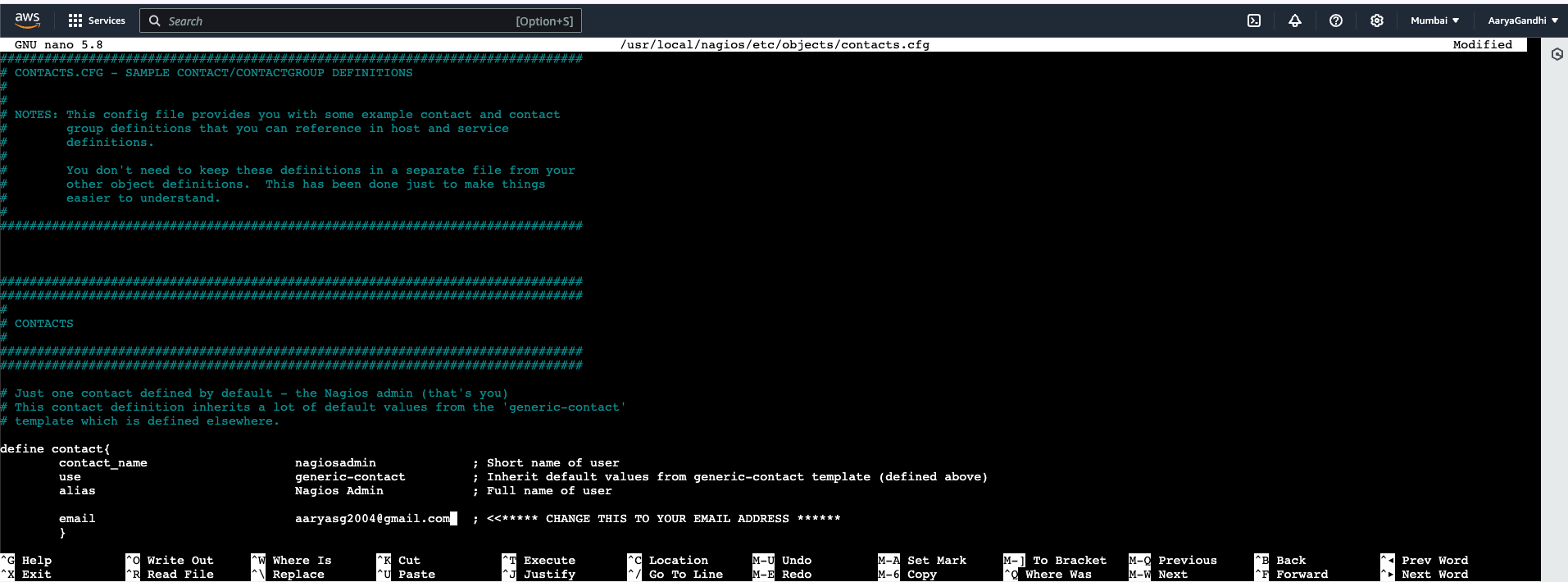
sudo make install

sudo make install-init

sudo make install-config

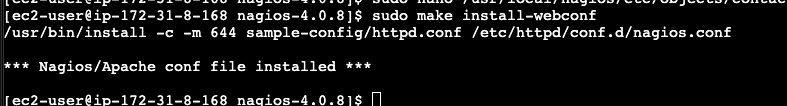
sudo make install-commandmode

14. Edit the config file and change the email address.



15. Configure the web interface.

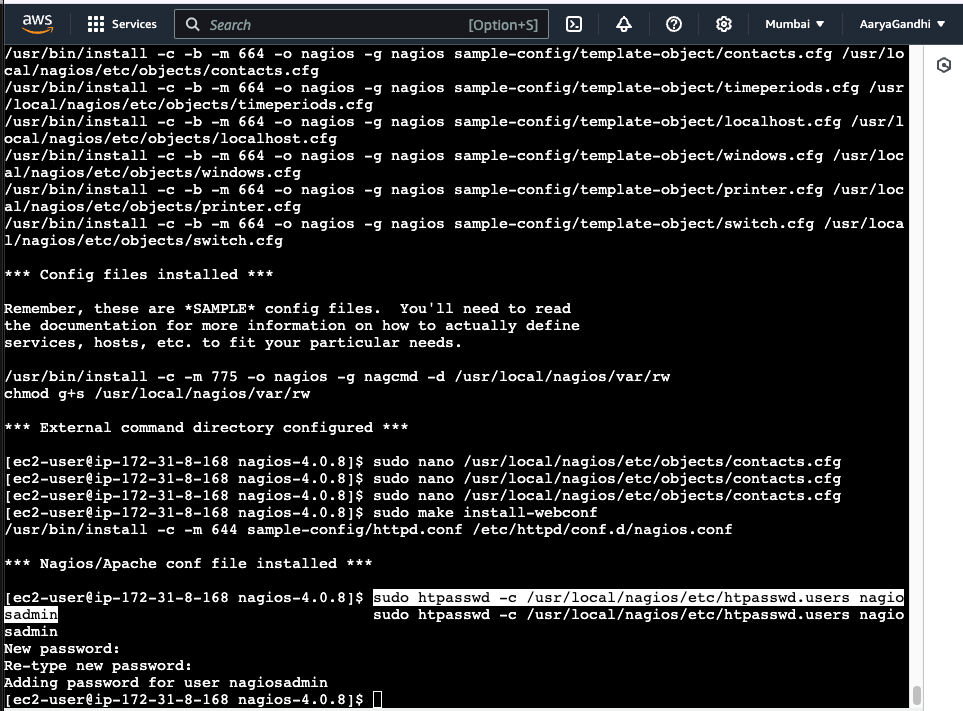
sudo make install-webconf



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



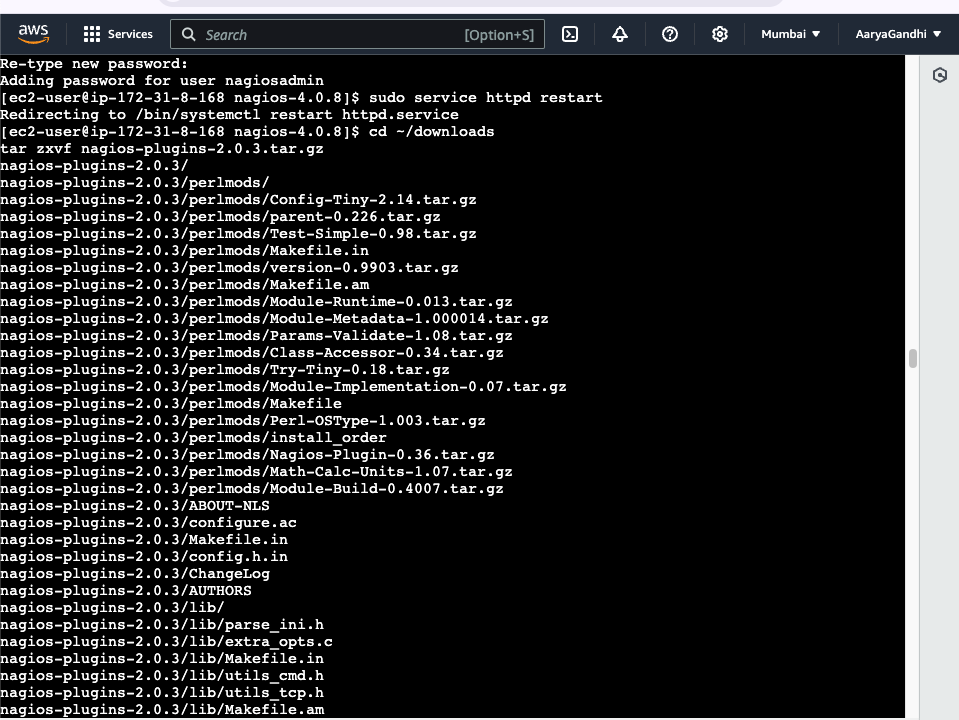
17. Restart Apache

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

20. Start Nagios

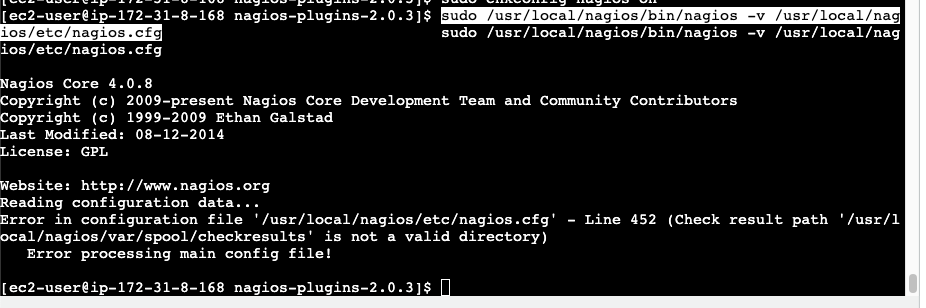
Add Nagios to the list of system services

sudo chkconfig --add nagios

sudo chkconfig nagios on

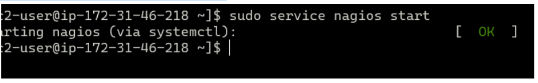
Verify the sample configuration files

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



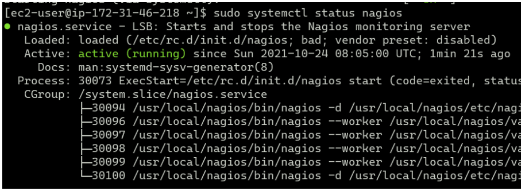
If there are no errors, you can go ahead and start Nagios.

sudo service nagios start

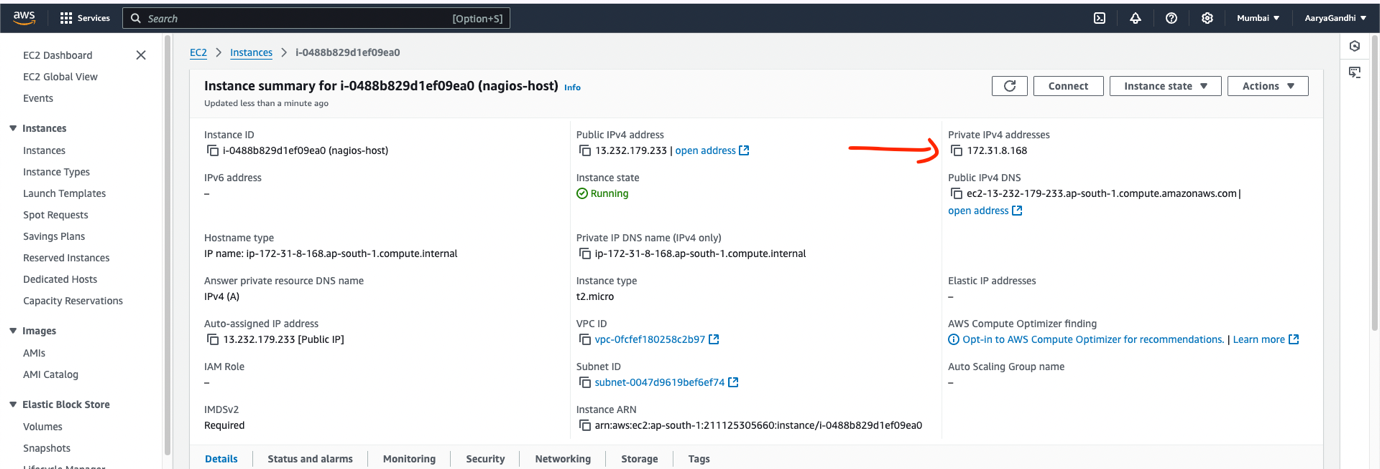


21. Check the status of Nagios

sudo systemctl status nagios

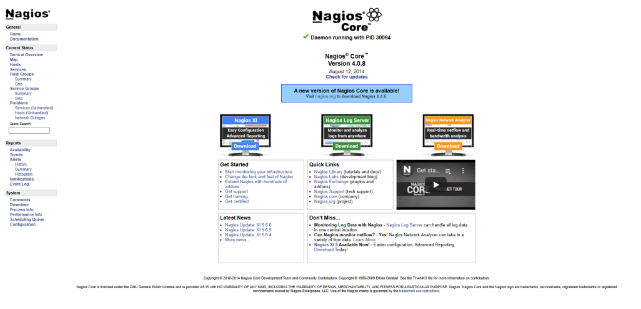


22. Go back to EC2 Console and copy the Public IP address of this instance



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

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



**Conclusion:**

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

machine.