

Name: Komal Singh

Div: D15B

Roll No: 60

Experiment: 9

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 We Need Nagios tool?

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- Helps you to plan for infrastructure upgrades before outdated systems create failures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

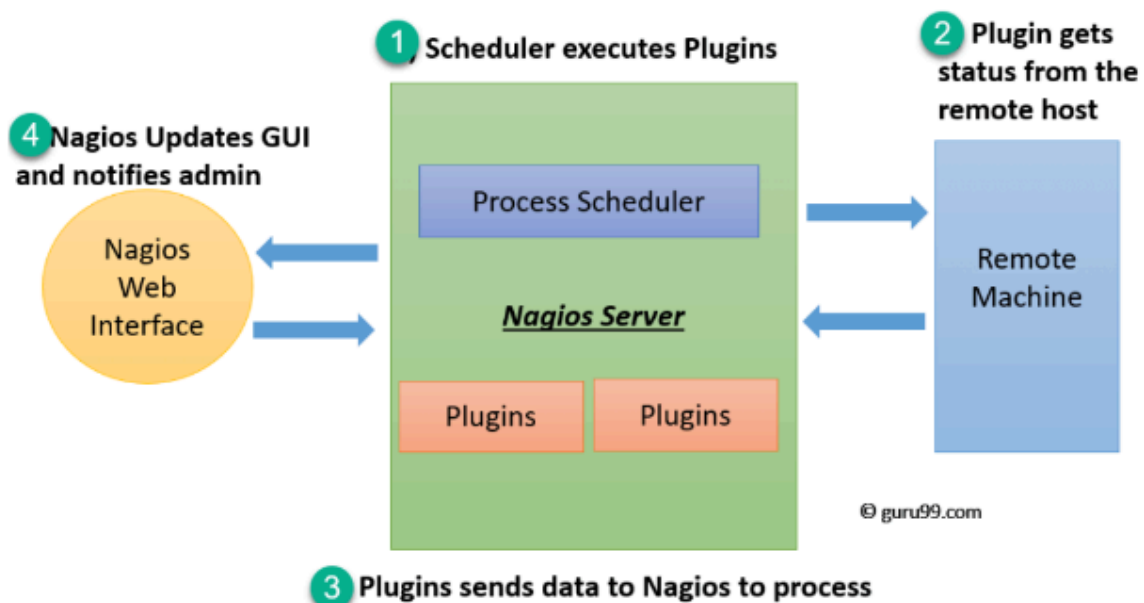
Following are the important features of Nagios monitoring tool:

- 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 do check that 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 runs 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.



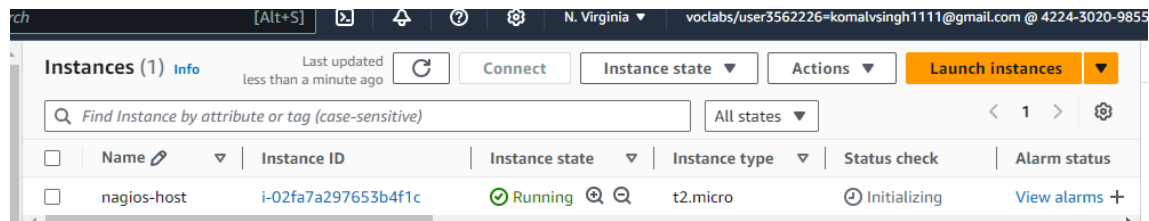
1. The scheduler is a component of the server part of Nagios. It sends a signal to execute the plugins at the remote host.
2. The plugin gets the status from the remote host
3. The plugin sends the data to the process scheduler
4. The process scheduler updates the GUI and notifications are sent to admins.

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.

The screenshot shows the 'Inbound rules (7)' section of the AWS Management Console. It displays a table of inbound rules for a security group. The rules are as follows:

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-08638a08b767b9...	IPv4	All ICMP - IPv6	IPv6 ICMP	All
-	sgr-051b14cb1d9d19e9f	IPv4	HTTPS	TCP	443
-	sgr-028eab2c8a63a0aca	IPv4	All traffic	All	All
-	sgr-00d991e6cc1386f7d	IPv4	All ICMP - IPv4	ICMP	All
-	sgr-0f1fa4bc6497acb39	IPv4	HTTP	TCP	80

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
```

```
[ec2-user@ip-172-31-44-218 ~]$ sudo yum update
sudo yum install httpd php
sudo yum install gcc glibc glibc-common
sudo yum install gd gd-devel
Last metadata expiration check: 0:00:20 ago on Sat Oct 12 09:32:21 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 0:00:20 ago on Sat Oct 12 09:32:21 2024.
Dependencies resolved.
```

Package	Architecture	Version
Installing:		
httpd	x86_64	2.4.62-1.amzn2023
php8.3	x86_64	8.3.10-1.amzn2023.0.1
Installing dependencies:		
apr	x86_64	1.7.2-2.amzn2023.0.2

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
```

```

complete:
[ec2-user@ip-172-31-44-218 ~]$ sudo adduser -m nagios
sudo passwd nagios
Changing password for user nagios.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-44-218 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-44-218 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-44-218 ~]$ mkdir ~/downloads
cd ~/downloads

```

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
```

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
```

```

[ec2-user@ip-172-31-44-218 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2024-10-12 09:37:11-- 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.tar.gz      100%[=====] 2.54M  5.51MB/s
2024-10-12 09:37:11 (5.51 MB/s) - 'nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]

[ec2-user@ip-172-31-44-218 downloads]$ tar zxvf nagios-4.0.8.tar.gz
tar (child): nagios-4.0.8.tar.gz: Cannot open: No such file or directory

```

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
```

```

tar: Error is not recoverable: exiting now
[ec2-user@ip-172-31-44-218 downloads]$ tar zxvf nagios-plugins-2.0.3.tar.gz
nagios-plugins-2.0.3/
nagios-plugins-2.0.3/perlmods/
nagios-plugins-2.0.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.0.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.0.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.in
nagios-plugins-2.0.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.am
nagios-plugins-2.0.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.0.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.0.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.0.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile
nagios-plugins-2.0.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.0.3/perlmods/install_order
nagios-plugins-2.0.3/perlmods/Nagios-Plugin-0.36.tar.gz
nagios-plugins-2.0.3/perlmods/Math-Calc-Units-1.07.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Build-0.4007.tar.gz
nagios-plugins-2.0.3/ABOUT-NLS

```

12. Compile the source code.

make all

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-172-31-44-218 downloads]$ cd nagios-4.0.8
./configure --with-command-group=nagcmd
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
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
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes

```

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

sudo nano /usr/local/nagios/etc/objects/contacts.cfg

```
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/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 nekmodes.o nekmodes.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nerd.o nerd.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_vpccool_list',
    inlined from 'get_worker' at workers.c:224:12:
workers.c:209:17: warning: '%s' directive argument is null [-Wformat-overflow=]
  209 |         log_debug_info(DEBUGL_CHECKS, 1, "Found specialized worker(s) for '%s'", (slash && *slash != '/') ? slash : cmd_name);
      |         ^~~~~~
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o checks.o checks.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o config.o config.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o commands.o commands.c
commands.c: In function 'process_passive_service_check':
commands.c:2247:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
 2247 |         cr.source = command_worker.source_name;
      |         ^
commands.c: In function 'process_passive_host_check':
commands.c:2339:19: warning: assignment discards 'const' qualifier from pointer target type [-Wdiscarded-qualifiers]
 2339 |         cr.source = command_worker.source_name;
      |         ^
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
```

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

```
make: *** [Makefile:72: all] Error 2
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ 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-4.0.8/base'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagios /usr/local/nagios/bin
/usr/bin/install -c -m 774 -o nagios -g nagios nagiosstats /usr/local/nagios/bin
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make strip-post-install
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/base'
/usr/bin/strip /usr/local/nagios/bin/nagios
/usr/bin/strip /usr/local/nagios/bin/nagiosstats
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.0.8/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.0.8/cgi'
```

```
GNU nano 5.8 /usr/local/nagios/etc/objects/contacts.cfg
#
# =====
# CONTACTS
# =====
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the 'generic-contact'
# template which is defined elsewhere.

define contact{
    contact_name    nagiosadmin           ; Short name of user
    use              generic-contact       ; Inherit default values from generic-contact template (defined above)
    alias            Nagios Admin          ; Full name of user

    email           komalvsingh111@gmail.com ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>>
}

^G Help      ^O Write Out  ^W Where Is   ^C Cut         ^E Execute    ^L Location   ^U Undo       ^M Set Mark   ^I To Bracket  ^P Previous
^X Exit      ^R Read File  ^N Replace    ^V Paste      ^J Justify    ^_ Go To Line  ^F Redo       ^K Copy       ^O Where Was  ^N Next
```

```
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf

*** Nagios/Apache conf file installed ***

[ec2-user@ip-172-31-44-218 nagios-4.0.8]$
```

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
```

```
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$
```

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
```

```
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-44-218 nagios-4.0.8]$
```

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
```

```

[ec2-user@ip-172-31-44-218 nagios-4.0.8]# sudo systemctl restart httpd.service
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-44-218 nagios-4.0.8]# cd ~/downloads
tar zxvf nagios-plugins-2.0.3.tar.gz
nagios-plugins-2.0.3/
nagios-plugins-2.0.3/perlmods/
nagios-plugins-2.0.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.0.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.0.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.in
nagios-plugins-2.0.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.am
nagios-plugins-2.0.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.0.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.0.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.0.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile
nagios-plugins-2.0.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.0.3/perlmods/install_order
nagios-plugins-2.0.3/perlmods/Nagios-Plugin-0.36.tar.gz
nagios-plugins-2.0.3/perlmods/Math-Calc-Units-1.07.tar.gz

```

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

```

[ec2-user@ip-172-31-44-218 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
checking host system type... x86_64-unknown-linux-gnu
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
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 for style of include used by make... GNU
checking dependency style of gcc... gcc3
checking how to run the C preprocessor... gcc -E

```

```

[ec2-user@ip-172-31-44-218 nagios-plugins-2.0.3]$ sudo chkconfig --add nagios
sudo chkconfig nagios on
[ec2-user@ip-172-31-44-218 nagios-plugins-2.0.3]$

```

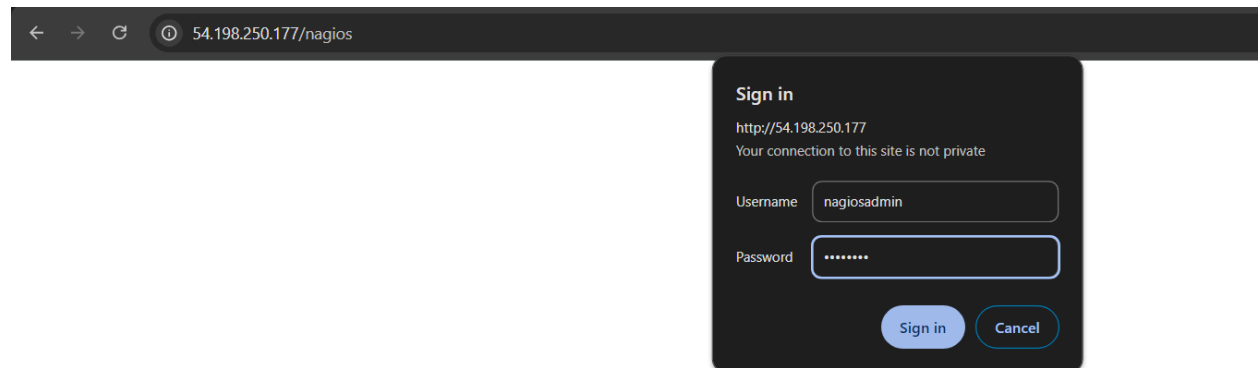


```
Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-44-218 nagios-plugins-2.0.3]$ sudo service nagios start
Reloading systemd: [ OK ]
Starting nagios (via systemctl): [ OK ]
[ec2-user@ip-172-31-44-218 nagios-plugins-2.0.3]$
```

```
Starting nagios (via systemctl): [ OK ]
[ec2-user@ip-172-31-44-218 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; generated)
   Active: active (running) since Sat 2024-10-12 09:59:46 UTC; 51s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 66468 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
    Tasks: 6 (limit: 1112)
   Memory: 2.1M
      CPU: 51ms
  CGroup: /system.slice/nagios.service
          └─66490 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─66492 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
                └─66493 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
                   └─66494 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
                      └─66495 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.gh
                         └─66496 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

oct 12 09:59:46 ip-172-31-44-218.ec2.internal nagios[66490]: wproc: Registry request: name=Core Worker 66493;pid=66493
oct 12 09:59:46 ip-172-31-44-218.ec2.internal nagios[66490]: wproc: Registry request: name=Core Worker 66492;pid=66492
oct 12 09:59:46 ip-172-31-44-218.ec2.internal nagios[66490]: Warning: Could not open object cache file '/usr/local/nagios/var/objects.cache' for writing!
oct 12 09:59:46 ip-172-31-44-218.ec2.internal nagios[66490]: Error: Unable to create temp file '/usr/local/nagios/var/nagios.tmp71290s' for writing status data
oct 12 09:59:46 ip-172-31-44-218.ec2.internal nagios[66490]: Successfully launched command file worker with pid 66496
oct 12 09:59:55 ip-172-31-44-218.ec2.internal nagios[66490]: Error: Unable to create temp file '/usr/local/nagios/var/nagios.tmp8hbMOK' for writing status data
```

23. Open up your browser and look for http://<your_public_ip_address>/nagios



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

