

ADVANCE DEVOPS EXP-9

ANSH SARFARE

D15A/50

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

Step-1: Login to your AWS account Personal / Academy. Click on EC2 instance then click on Create Security Group. Give the name as Nagios and any description and add the following inbound rules.

The screenshot shows the AWS Management Console for a Security Group named 'Nagios'. The 'Inbound rules' tab is selected, displaying a table of 7 inbound rules. The rules are as follows:

Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
-	sgr-030b9c5c0081d88...	IPv4	HTTPS	TCP	443	0.0.0.0/0	-
-	sgr-0d61c2da4246c9df1	IPv4	Custom TCP	TCP	5666	0.0.0.0/0	-
-	sgr-0355ee35073d3a8...	IPv4	All traffic	All	All	0.0.0.0/0	-
-	sgr-0b6f61aa6f97649fb	IPv4	SSH	TCP	22	0.0.0.0/0	-
-	sgr-0ca4370fb26bf9152	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0	-
-	sgr-0ad59a68912427efc	IPv6	All ICMP - IPv6	IPv6 ICMP	All	::/0	-
-	sgr-059c3b71c6e905bfc	IPv4	HTTP	TCP	80	0.0.0.0/0	-

Step 2: Now Create a new EC2 instance. Name: Nagios-host ,AMI: Amazon Linux, Instance Type: t2.micro.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Name and tags' section has 'Nagios-host' entered. The 'Application and OS Images (Amazon Machine Image)' section shows 'Amazon Linux 2023.5.2' selected. The 'Summary' section on the right shows the configuration: 1 instance, Amazon Linux 2023.5.2 AMI, t2.micro instance type, new security group, and 1 volume (8 GiB). A 'Free tier' notice is displayed, stating that the first year includes 750 hours of t2.micro usage. The 'Launch instance' button is highlighted in orange.

For Key pair : Click on create key and make key of type RSA with extension .pem . Key will be downloaded to your local machine and select the **Existing Security Group** and select the Security Group we have created in Step 1.

Instance type

t2.micro

Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Nagios

Create new key pair

▼ Network settings Info

Edit

Network Info

vpc-07294e1d226906dc2

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

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

Nagios sg-0ac9e4ec153c969df X
VPC: vpc-07294e1d226906dc2

Compare security group rules

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)

Amazon Linux 2023.5.2...read more
ami-0ebfd941bbafe70c6

Virtual server type (instance type)

t2.micro

Firewall (security group)

Nagios

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Review commands

[Alt+S]

Instances (1/1) Info

Last updated 1 minute ago

Connect

Instance state

Actions

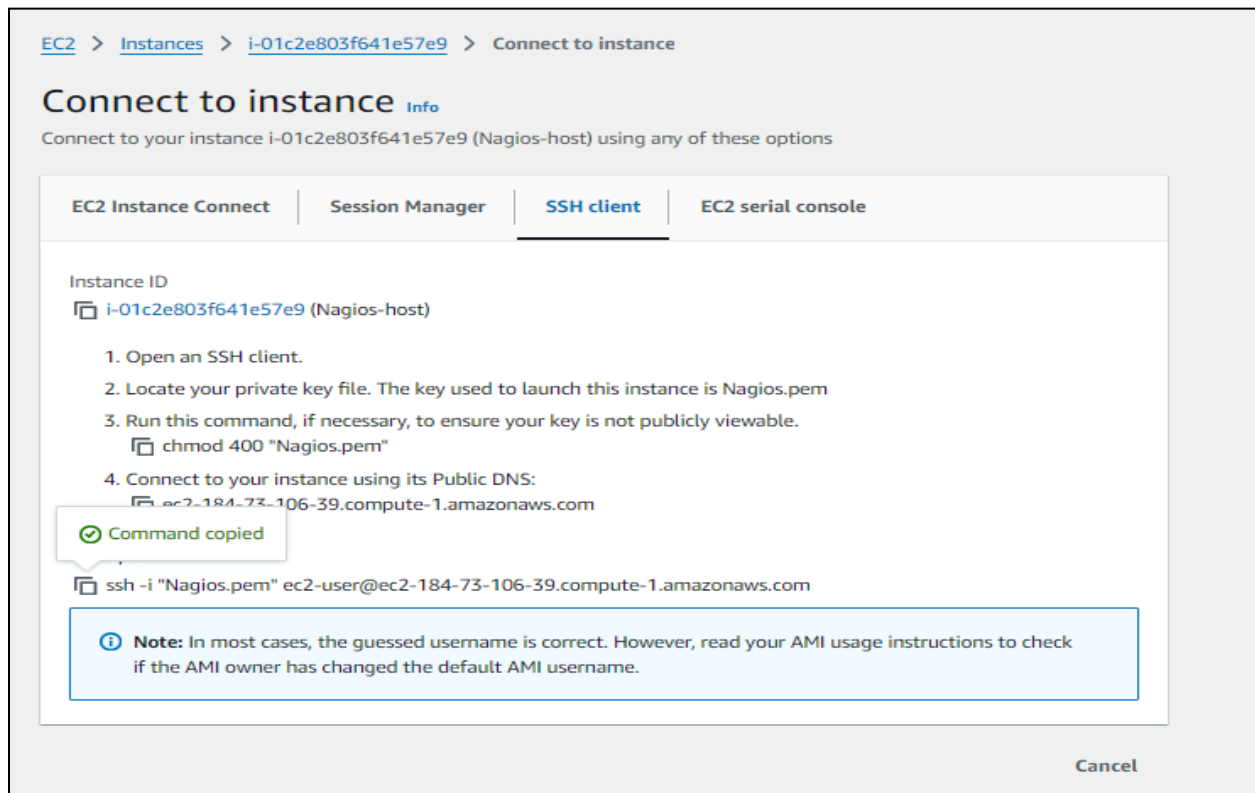
Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs	Monitoring	Security g
Nagios-host	i-01c2e803f641e57e9	Running	t2.micro	Initializing	View alarms +	us-east-1d	ec2-184-73-106-39.co...	184.73.106.39	-	-	disabled	Nagios

Step 3: Now After creating the EC2 Instance click on connect and then copy the command which is given as example in the SSH Client section .



Successfully connected to the instance.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Ansh> cd Desktop
PS C:\Users\Ansh\Desktop> cd Nagios
PS C:\Users\Ansh\Desktop\Nagios> ssh -i "Nagios.pem" ec2-user@ec2-184-73-106-39.compute-1.amazonaws.com
The authenticity of host 'ec2-184-73-106-39.compute-1.amazonaws.com (184.73.106.39)' can't be established.
ED25519 key fingerprint is SHA256:i7NM6Wl0yzJAd4YHlwTc+dDHM9igux+4/N/7+Lq4xU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-184-73-106-39.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

#
~\_#####_ Amazon Linux 2023
NN\_#####\
NN\_####|
NN\_#\_/___ https://aws.amazon.com/linux/amazon-linux-2023
NN      V~' '->
   NNN
   NN.-.-/
   _/_/_/
   _/m/'

[ec2-user@ip-172-31-90-152 ~]$ |
```

Step 4: Now Run the following command to make a new user.- sudo adduser -m nagios
sudo passwd nagios

```
[ec2-user@ip-172-31-90-152 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-90-152 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-90-152 ~]$ |
```

Step 5: Now Run the following command to make a new user group.
sudo groupadd nagcmd ,
sudo usermod -a -G nagcmd nagios,
sudo usermod -a -G nagcmd apache

If apache is not installed on your system:

```
[ec2-user@ip-172-31-90-152 ~]$ sudo yum update -y
Last metadata expiration check: 0:13:57 ago on Mon Sep 30 17:57:12 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-90-152 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:15:05 ago on Mon Sep 30 17:57:12 2024.
Dependencies resolved.
=====
Package                                Architecture      Version
=====
Installing:
httpd                                   x86_64            2.4.62-1.amzn2023
Installing dependencies:
apr                                     x86_64            1.7.2-2.amzn2023.0.2
apr-util                               x86_64            1.6.3-1.amzn2023.0.1
=====
```

```
[ec2-user@ip-172-31-90-152 ~]$ sudo groupadd nagcmd
groupadd: group 'nagcmd' already exists
[ec2-user@ip-172-31-90-152 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-90-152 ~]$ sudo usermod -a -G nagcmd httpd
usermod: user 'httpd' does not exist
[ec2-user@ip-172-31-90-152 ~]$ ps aux | grep apache
apache    22421  0.0  0.4 17020 4580 ?        S   18:12   0:00 /usr/sbin/httpd -DFOREGROUND
apache    22426  0.0  0.7 1084984 7560 ?        Sl  18:12   0:00 /usr/sbin/httpd -DFOREGROUND
apache    22427  0.0  0.7 1084984 7560 ?        Sl  18:12   0:00 /usr/sbin/httpd -DFOREGROUND
apache    22428  0.0  0.7 1248888 7560 ?        Sl  18:12   0:00 /usr/sbin/httpd -DFOREGROUND
ec2-user  26483  0.0  0.2 222312 2064 pts/0    S+  18:14   0:00 grep --color=auto apache
[ec2-user@ip-172-31-90-152 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-90-152 ~]$ sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-90-152 ~]$ |
```

Step 6: Now make a new directory and go to that directory. `mkdir ~/downloads`
`cd ~/downloads`

```
[ec2-user@ip-172-31-90-152 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-90-152 downloads]$ |
```

Step 7: Now to download the Nagios 4.5.5 and Nagios-plugins 2.4.11 run the following commands respectively.

wget <https://go.nagios.org/l/975333/2024-09-17/6kqcx>

```
[ec2-user@ip-172-31-90-152 downloads]$ wget https://go.nagios.org/l/975333/2024-09-17/6kqcx
--2024-09-30 18:18:59-- https://go.nagios.org/l/975333/2024-09-17/6kqcx
Resolving go.nagios.org (go.nagios.org)... 52.54.96.194, 3.92.120.28, 3.215.172.219, ...
Connecting to go.nagios.org (go.nagios.org)|52.54.96.194|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: http://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz?utm_source=Nagios.org&utm_content=Download+&pi_content=1e9662c93afb2ed6bd2e3f3cc38771a7f01125e969f2a75b0e2254439d4a81d8 [following]
--2024-09-30 18:18:59-- http://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz?utm_source=Nagios.org&utm_content=Download+&pi_content=1e9662c93afb2ed6bd2e3f3cc38771a7f01125e969f2a75b0e2254439d4a81d8
Resolving assets.nagios.com (assets.nagios.com)... 45.79.49.120, 2600:3c00::f03c:92ff:fe77:45ce
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz?utm_source=Nagios.org&utm_content=Download+&pi_content=1e9662c93afb2ed6bd2e3f3cc38771a7f01125e969f2a75b0e2254439d4a81d8 [following]
--2024-09-30 18:18:59-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz?utm_source=Nagios.org&utm_content=Download+&pi_content=1e9662c93afb2ed6bd2e3f3cc38771a7f01125e969f2a75b0e2254439d4a81d8
Connecting to assets.nagios.com (assets.nagios.com)|45.79.49.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2065473 (2.0M) [application/x-gzip]
Saving to: '6kqcx'

6kqcx                               100%[=====] 1.97M  8.62MB/s   in 0.2s

2024-09-30 18:18:59 (8.62 MB/s) - '6kqcx' saved [2065473/2065473]
```

wget <https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz>

```
[ec2-user@ip-172-31-90-152 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-09-30 18:20:08-- https://nagios-plugins.org/download/nagios-plugins-2.4.11.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|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2753049 (2.6M) [application/x-gzip]
Saving to: 'nagios-plugins-2.4.11.tar.gz'

nagios-plugins-2.4.11.tar.gz       100%[=====]

2024-09-30 18:20:09 (10.2 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]
```

Step 8: Now to extract the files from the downloaded Nagios 4.5.5 run the following command. `tar zxvf 6kqcx`

```
[ec2-user@ip-172-31-90-152 downloads]$ tar zxvf 6kqcx
nagios-4.5.5/
nagios-4.5.5/.github/
nagios-4.5.5/.github/workflows/
nagios-4.5.5/.github/workflows/test.yml
nagios-4.5.5/.gitignore
nagios-4.5.5/CONTRIBUTING.md
nagios-4.5.5/Changelog
nagios-4.5.5/INSTALLING
nagios-4.5.5/LLEGAL
nagios-4.5.5/LICENSE
nagios-4.5.5/Makefile.in
nagios-4.5.5/README.md
nagios-4.5.5/THANKS
nagios-4.5.5/UPGRADING
nagios-4.5.5/aclocal.m4
nagios-4.5.5/autoconf-macros/
nagios-4.5.5/autoconf-macros/.gitignore
nagios-4.5.5/autoconf-macros/CHANGELOG.md
nagios-4.5.5/autoconf-macros/LICENSE
nagios-4.5.5/autoconf-macros/LICENSE.md
nagios-4.5.5/autoconf-macros/README.md
nagios-4.5.5/autoconf-macros/add_group_user
nagios-4.5.5/autoconf-macros/ax_nagios_get_distrib
```

Step 9: Now change the directory to nagios-4.5.5

```
[ec2-user@ip-172-31-90-152 downloads]$ cd nagios-4.5.5
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |
```

Step 10: Now run the following command to configure `./configure`

`--with-command-group=nagcmd`.

In this step you can get the error, it might be due to gcc is not installed in your system so to install it run the following command: **`sudo yum groupinstall "Development`**

`Tools" -y`

```
[ec2-user@ip-172-31-90-152 downloads]$ cd nagios-4.5.5
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ ./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 the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for stdio.h... yes
checking for stdlib.h... yes
```

At the end we have found the error of cannot find ssl header .

```
checking for Kerberos include files... configure: WARNING: could not find include files
checking for pkg-config... pkg-config
checking for SSL headers... configure: error: Cannot find ssl headers
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |
```

So run following command to install ssl. sudo yum install openssl-devel

```
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo yum install openssl-devel
Last metadata expiration check: 0:39:47 ago on Mon Sep 30 17:57:12 2024.
Dependencies resolved.
=====
Package                                Architecture                          Version
=====
Installing:
openssl-devel                          x86_64                                1:3.0.8-1.amzn2023.0.14

Transaction Summary
=====
Install 1 Package

Total download size: 3.0 M
Installed size: 4.7 M
Is this ok [y/N]: y
Downloading Packages:
openssl-devel-3.0.8-1.amzn2023.0.14.x86_64.rpm
-----
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      : 
  Installing     : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64
  Running scriptlet: openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64
  Verifying      : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64

Installed:
  openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64

Complete!
```

Now rerun the command ./configure--with-command-group=nagcmd

```
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ ./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 the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for stdio.h... yes
checking for stdlib.h... yes
```

```

IOBroker Method:  epoll

Web Interface Options:
-----
                HTML URL:  http://localhost/nagios/
                CGI URL:   http://localhost/nagios/cgi-bin/
Traceroute (used by WAP): /usr/bin/traceroute

Review the options above for accuracy.  If they look okay,
type 'make all' to compile the main program and CGIs.

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ make all
cd ./base && make

*** Support Notes ****

If you have questions about configuring or running Nagios,
please make sure that you:

    - Look at the sample config files
    - Read the documentation on the Nagios Library at:
      https://library.nagios.com

before you post a question to one of the mailing lists.
Also make sure to include pertinent information that could
help others help you.  This might include:

    - What version of Nagios you are using
    - What version of the plugins you are using
    - Relevant snippets from your config files
    - Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:

    https://support.nagios.com

*****

Enjoy.

```

Step 11: Now run the following commands to setup the Nagios. sudo make install

```

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/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 nagiosstats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -s -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin
done

```



```

*** Main program, CGIs and HTML files installed ***

You can continue with installing Nagios as follows (type 'make'
without any arguments for a list of all possible options):

    make install-init
        - This installs the init script in /lib/systemd/system

    make install-commandmode
        - This installs and configures permissions on the
          directory for holding the external command file

    make install-config
        - This installs sample config files in /usr/local/nagios/etc

make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5'

```

sudo make install-init

```

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |

```

sudo make install-config

```

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg

*** Config files installed ***

Remember, these are *SAMPLE* config files. You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.

```

sudo make install-webconf

```

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/httpd/conf.d/nagios.conf
if [ 0 -eq 1 ]; then \
    ln -s /etc/httpd/conf.d/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \
fi

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

[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |

```

`sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin`

```
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |
```

Now to restart the httpd service run the following command. `sudo service httpd restart`

```
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ |
```

Step 12: Now to extract the files from the downloaded Nagios plugin 2.4.11 run the following command first change the directory.

`cd ~/downloads`

`tar zxvf nagios-plugins-2.4.11.tar.gz`

```
[ec2-user@ip-172-31-90-152 nagios-4.5.5]$ cd ~/downloads
[ec2-user@ip-172-31-90-152 downloads]$ tar zxvf nagios-plugins-2.4.11.tar.gz
nagios-plugins-2.4.11/
nagios-plugins-2.4.11/build-aux/
nagios-plugins-2.4.11/build-aux/compile
nagios-plugins-2.4.11/build-aux/config.guess
nagios-plugins-2.4.11/build-aux/config.rpath
nagios-plugins-2.4.11/build-aux/config.sub
nagios-plugins-2.4.11/build-aux/install-sh
nagios-plugins-2.4.11/build-aux/ltmain.sh
nagios-plugins-2.4.11/build-aux/missing
nagios-plugins-2.4.11/build-aux/mkinstalldirs
nagios-plugins-2.4.11/build-aux/depcomp
nagios-plugins-2.4.11/build-aux/snippet/
nagios-plugins-2.4.11/build-aux/snippet/_Noreturn.h
nagios-plugins-2.4.11/build-aux/snippet/arg-nonnull.h
nagios-plugins-2.4.11/build-aux/snippet/c++defs.h
nagios-plugins-2.4.11/build-aux/snippet/warn-on-use.h
nagios-plugins-2.4.11/build-aux/test-driver
nagios-plugins-2.4.11/config_test/
```

Step 13: Now change the directory to nagios-plugins-2.4.11 and run the config command to configure.

`cd nagios-plugins-2.4.11`

`./configure--with-nagios-user=nagios--with-nagios-group=nagios`

```
[ec2-user@ip-172-31-90-152 downloads]$ cd nagios-plugins-2.4.11
[ec2-user@ip-172-31-90-152 nagios-plugins-2.4.11]$ ./configure --with-nagios-user=nagios --with-nagios-group=nagios
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 make supports nested variables... yes
checking whether to enable maintainer-specific portions of Makefiles... yes
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
```

Step 14: Run the following commands to check nagios and start it.

`sudo chkconfig--add nagios`

```
[ec2-user@ip-172-31-90-152 nagios-plugins-2.4.11]$ sudo chkconfig --add nagios
sudo chkconfig nagios on
error reading information on service nagios: No such file or directory
Note: Forwarding request to 'systemctl enable nagios.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.
[ec2-user@ip-172-31-90-152 nagios-plugins-2.4.11]$
```

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

```
[ec2-user@ip-172-31-90-152 nagios-plugins-2.4.11]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.5.5
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-09-17
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 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 1 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
```

`cd`

`sudo service nagios start`

```
[ec2-user@ip-172-31-90-152 nagios-plugins-2.4.11]$ cd
[ec2-user@ip-172-31-90-152 ~]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-90-152 ~]$ |
```

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

```
[ec2-user@ip-172-31-90-152 ~]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.5.5
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-09-17
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 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 1 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
```

sudo systemctl restart nagios

sudo systemctl status nagios

```

bash: d: Command not found
[ec2-user@ip-172-31-90-152 nagios-plugins-2.3.3]$ cd
[ec2-user@ip-172-31-90-152 ~]$ sudo systemctl restart nagios
[ec2-user@ip-172-31-90-152 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.5
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-09-30 19:41:36 UTC; 7s ago
     Docs: https://www.nagios.org/documentation
   Process: 80238 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 80239 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 80240 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 4.0M
        CPU: 15ms
    CGroup: /system.slice/nagios.service
           └─80240 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─80241 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─80242 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                 └─80243 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─80244 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                     └─80245 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: core query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: echo service query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: qh: help for the query handler registered
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Successfully registered manager as @wproc with query handler
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80244;pid=80244
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80243;pid=80243
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80242;pid=80242
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: wproc: Registry request: name=Core Worker 80241;pid=80241
Sep 30 19:41:36 ip-172-31-90-152.ec2.internal nagios[80240]: Successfully launched command file worker with pid 80245

```

Step 15: We can see we have successfully launched the Nagios now . Open <http://nagios/> here it is <http://44.202.108.37/nagios> we can see the running web page of nagios.

The screenshot shows the Nagios Core web interface in a browser. The URL bar shows "Not secure 44.202.108.37/nagios/". The interface has a left sidebar with navigation links: General (Home, Documentation), Current Status (Tactical Overview, Map (Legacy), Hosts, Services, Host Groups, Service Groups, Problems), Reports (Availability, Trends (Legacy), Alerts, History, Notifications, Event Log), and System (Comments, Downtime, Process Info, Performance Info, Scheduling Queue, Configuration). The main content area displays the Nagios Core logo and the status "Daemon running with PID 68654". Below this, it shows "Nagios® Core™ Version 4.4.6" and "April 28, 2020". A blue banner announces "A new version of Nagios Core is available! Visit nagios.org to download Nagios 4.5.5." There are also sections for "Get Started", "Quick Links", "Latest News", and "Don't Miss...".