

## Adv.DevOps Exp 09

Name- Ishika Devare RollNo- 14 Batch- A

Adv. DevOps  
Experiment No. 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 internet. Nagios is used for continuous monitoring of systems, applications, service and business processes.

- Features of Nagios

- 1) Relatively scalable, Manageable & secure.
- 2) Informative and attractive web interfaces.
- 3) Automatically send alerts if condition changes.
- 4) Helps you detect network errors or server crashes.
- 5) Good log and database system.

- Why we need Nagios tool ?

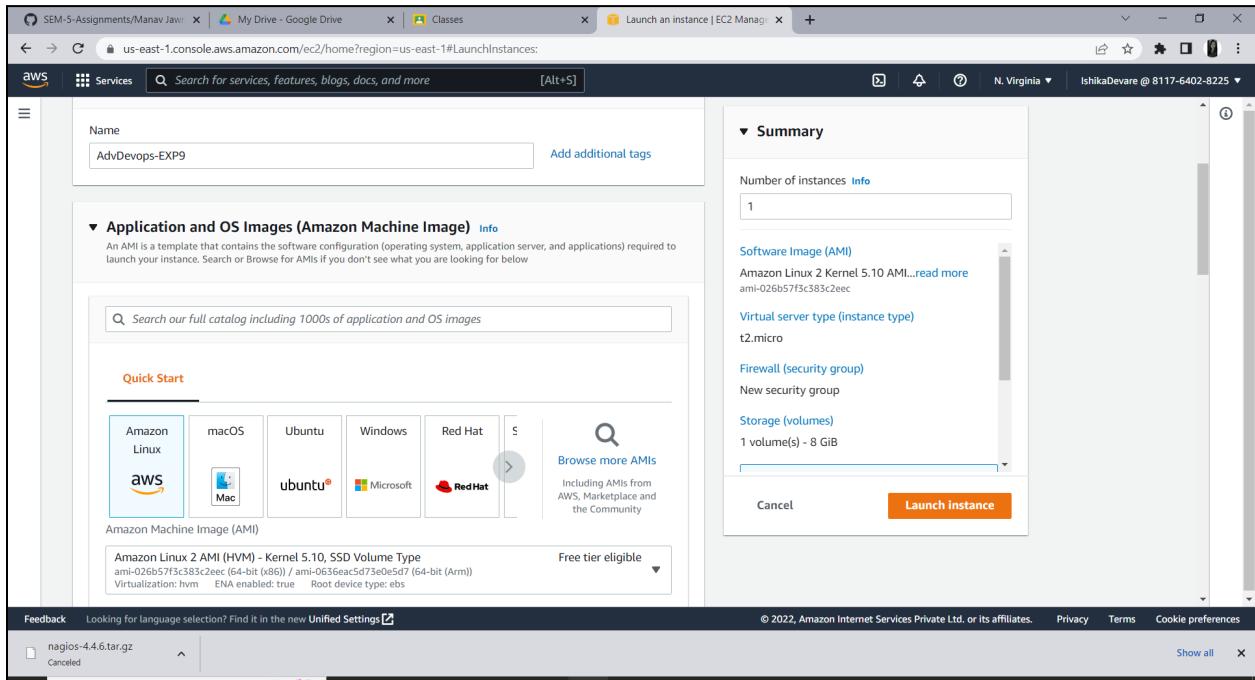
- 1) Detect all types of network or server issues
- 2) Active monitoring of your entire infrastructure & business processes.

 FOR EDUCATIONAL USE

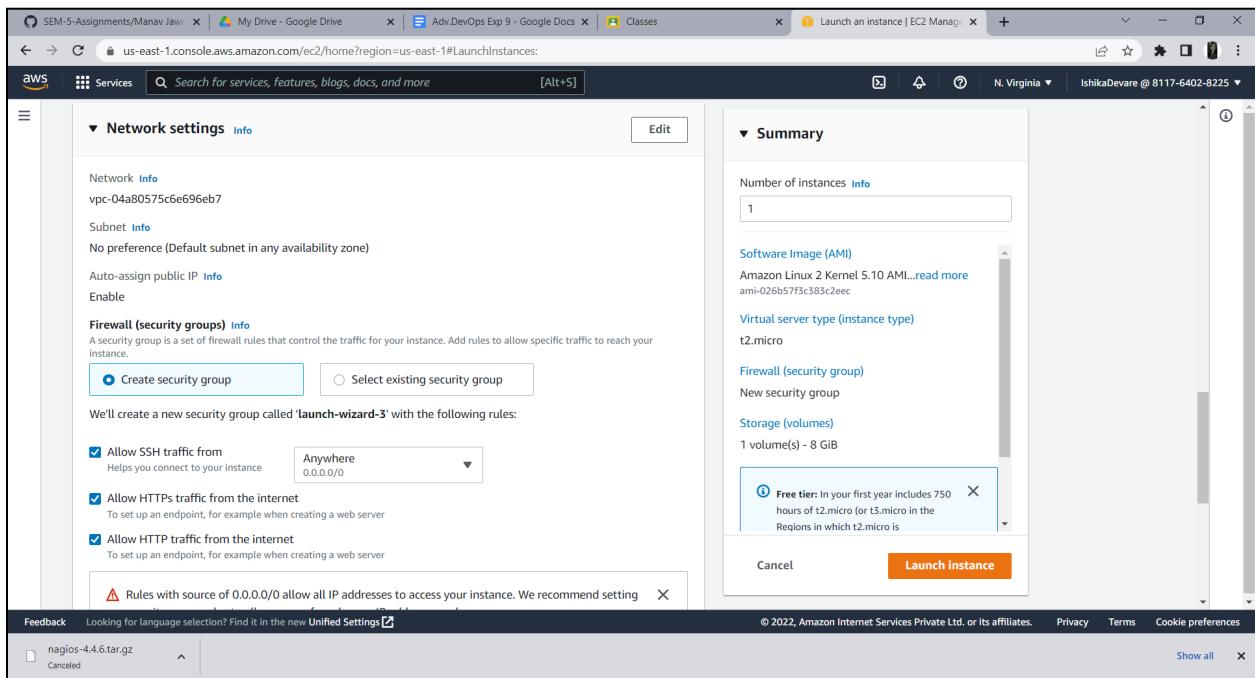
3)	Allows to monitor and troubleshoot server performance issues.
4)	You can maintain the security and availability of service.
5)	Automatically fix problems in panic situation.

## Implementation:

### Step 1: Create a Linux EC2 instance on AWS.



Create a new security group and Allow SSH , HTTP , HTTPS .



## Step 2: Install Prerequisite Software

Nagios requires the following packages are installed on your server prior to installing Nagios:

Install php using command **sudo yum install httpd php**

```
[ec2-user@ip-172-31-81-219 ~]$ sudo yum install httpd php
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.54-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: httpd-filesystem = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: system-logos-httpd for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: mod_http2 for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: httpd-filesystem for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: /etc/mime.types for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: libaprutil-1.so.0() (64bit) for package: httpd-2.4.54-1.amzn2.x86_64
---> Processing Dependency: libapr-1.so.0() (64bit) for package: httpd-2.4.54-1.amzn2.x86_64
---> Package php.x86_64 0:5.4.16-46.amzn2.0.2 will be installed
---> Processing Dependency: php-cli(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
---> Processing Dependency: php-common(x86-64) = 5.4.16-46.amzn2.0.2 for package: php-5.4.16-46.amzn2.0.2.x86_64
---> Running transaction check
---> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
```

Check php version using **php -v**

```
[ec2-user@ip-172-31-81-219 ~]$ php -v
PHP 5.4.16 (cli) (built: Oct 31 2019 18:34:05)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[ec2-user@ip-172-31-81-219 ~]$
```

Install gcc

```
[ec2-user@ip-172-31-81-219 ~]$ sudo yum install gcc glibc glibc-common
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Package glibc-2.26-60.amzn2.x86_64 already installed and latest version
Package glibc-common-2.26-60.amzn2.x86_64 already installed and latest version
Resolving Dependencies
--> Running transaction check
----> Package gcc.x86_64 0:7.3.1-15.amzn2 will be installed
----> Processing Dependency: cpp = 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libasanitizer >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libquadmath >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libmpx >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libitm >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libcilkrtts >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libatomic >= 7.3.1-15.amzn2 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: glibc-devel >= 2.2.90-12 for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libmpfr.so.4()(64bit) for package: gcc-7.3.1-15.amzn2.x86_64
----> Processing Dependency: libmpc.so.3()(64bit) for package: gcc-7.3.1-15.amzn2.x86_64
----> Running transaction check
----> Package cpp.x86_64 0:7.3.1-15.amzn2 will be installed
----> Package glibc-devel.x86_64 0:2.26-60.amzn2 will be installed
----> Processing Dependency: glibc-headers = 2.26-60.amzn2 for package: glibc-devel-2.26-60.amzn2.x86_64
----> Processing Dependency: glibc-headers for package: glibc-devel-2.26-60.amzn2.x86_64
----> Package libatomic.x86_64 0:7.3.1-15.amzn2 will be installed
----> Package libcilkrtts.x86_64 0:7.3.1-15.amzn2 will be installed
```

Check gcc version using command **gcc –version**

```
[ec2-user@ip-172-31-81-219 ~]$ gcc --version
gcc (GCC) 7.3.1 20180712 (Red Hat 7.3.1-15)
Copyright (C) 2017 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

```
[ec2-user@ip-172-31-81-219 ~]$ █
```

Install gd using command **sudo yum install gd gd-devel**

```
[ec2-user@ip-172-31-81-219 ~]$ sudo yum install gd gd-devel
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
----> Package gd.x86_64 0:2.0.35-27.amzn2 will be installed
----> Processing Dependency: libfontconfig.so.1()(64bit) for package: gd-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libXpm.so.4()(64bit) for package: gd-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libX11.so.6()(64bit) for package: gd-2.0.35-27.amzn2.x86_64
----> Package gd-devel.x86_64 0:2.0.35-27.amzn2 will be installed
----> Processing Dependency: zlib-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libpng-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libjpeg-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libXpm-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: libX11-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: freetype-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Processing Dependency: fontconfig-devel for package: gd-devel-2.0.35-27.amzn2.x86_64
----> Running transaction check
----> Package fontconfig.x86_64 0:2.13.0-4.3.amzn2 will be installed
----> Processing Dependency: fontpackages-filesystem for package: fontconfig-2.13.0-4.3.amzn2.x86_64
```

**Step 3:** Create Account Information

You need to set up a Nagios user. Run the following commands:

```
sudo adduser -m nagios
```

```
sudo passwd nagios
```

```
[ec2-user@ip-172-31-81-219 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-81-219 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-81-219 ~]$ █
```

Type the new password twice.

```
sudo groupadd nagcmd
```

```
sudo usermod -a -G nagcmd nagios
```

```
sudo usermod -a -G nagcmd apache
```

```
[ec2-user@ip-172-31-81-219 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-81-219 ~]$ sudo usermod -a -G nagcmd nagios
[ec2-user@ip-172-31-81-219 ~]$ sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-81-219 ~]$ █
```

#### **Step 4:** Download Nagios Core and the Plugins

Create a directory for storing the downloads.

```
mkdir ~/downloads
```

```
cd ~/downloads
```

```
[ec2-user@ip-172-31-81-219 ~]$ mkdir ~/downloads
[ec2-user@ip-172-31-81-219 ~]$ ls
downloads
[ec2-user@ip-172-31-81-219 ~]$ cd ~/downloads
[ec2-user@ip-172-31-81-219 downloads]$ █
```

Now install nagios using command:

```
wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.8.tar.gz
```

```
[ec2-user@ip-172-31-81-219 downloads]$ wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.5.tar.gz
--2022-10-01 07:52:40-- http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-4.0.5.tar.gz
Resolving prdownloads.sourceforge.net (prdownloads.sourceforge.net)... 204.68.111.105
Connecting to prdownloads.sourceforge.net (prdownloads.sourceforge.net)|204.68.111.105|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.5/nagios-4.0.5.tar.gz [following]
--2022-10-01 07:52:40-- http://downloads.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.5/nagios-4.0.5.tar.gz
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 204.68.111.105
Reusing existing connection to prdownloads.sourceforge.net:80.
HTTP request sent, awaiting response... 302 Found
Location: http://master.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.5/nagios-4.0.5.tar.gz?viasf=1 [following]
--2022-10-01 07:52:40-- http://master.dl.sourceforge.net/project/nagios/nagios-4.x/nagios-4.0.5/nagios-4.0.5.tar.gz?viasf=1
Resolving master.dl.sourceforge.net (master.dl.sourceforge.net)... 216.105.38.12
Connecting to master.dl.sourceforge.net (master.dl.sourceforge.net)|216.105.38.12|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1793038 (1.7M) [application/x-gzip]
Saving to: 'nagios-4.0.5.tar.gz'

0% [=====] 7% [=====>]
53% [=====>] 1,793,038 3.49MB/s in 0.5s 100%[=====]

2022-10-01 07:52:43 (3.49 MB/s) - 'nagios-4.0.5.tar.gz' saved [1793038/1793038]

[ec2-user@ip-172-31-81-219 downloads]$
```

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

```
[ec2-user@ip-172-31-81-219 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2022-10-01 10:35:23--  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'

100%[=====] 2,659,772    7.17MB/s    in 0.4s

2022-10-01 10:35:23 (7.17 MB/s) - 'nagios-plugins-2.0.3.tar.gz' saved [2659772/2659772]

[ec2-user@ip-172-31-81-219 downloads]$ █
```

**Step 5:** Compile and Install Nagios, use the following command:

```
tar zxvf nagios-4.0.8.tar.gz
```

```
[ec2-user@ip-172-31-81-219 downloads]$ tar zxvf nagios-4.0.8.tar.gz
nagios-4.0.8/
nagios-4.0.8/.gitignore
nagios-4.0.8/Changelog
nagios-4.0.8/INSTALLING
nagios-4.0.8/LEGAL
nagios-4.0.8/LICENSE
nagios-4.0.8/Makefile.in
nagios-4.0.8/README
nagios-4.0.8/README.asciidoc
nagios-4.0.8/THANKS
nagios-4.0.8/UPGRADING
nagios-4.0.8/base/
nagios-4.0.8/base/.gitignore
nagios-4.0.8/base/Makefile.in
nagios-4.0.8/base/broker.c
nagios-4.0.8/base/checks.c
nagios-4.0.8/base/commands.c
nagios-4.0.8/base/config.c
nagios-4.0.8/base/events.c
nagios-4.0.8/base/flapping.c
nagios-4.0.8/base/logging.c
nagios-4.0.8/base/nagios.c
```

**cd nagios-plugins-4.0.8**

```
[ec2-user@ip-172-31-81-219 downloads]$ cd nagios-4.0.8
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ █
```

Run the configuration script with the name of the group which you have created in the above step.

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

```
[ec2-user@ip-172-31-81-219 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
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
```

Compile the Nagios source code.

## **make all**

```
*** 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:  
<http://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:

<http://support.nagios.com>

Install binaries, init script, sample config files and set permissions on the external command directory.

## **sudo make install**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo make install
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 nagiostats /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/nagiostats
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'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
```

## **sudo make install-init**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /etc/rc.d/init.d
/usr/bin/install -c -m 755 -o root -g root daemon-init /etc/rc.d/init.d/nagios

*** Init script installed ***

[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

## **sudo make install-config**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ 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
```

## **sudo make install-commandmode**

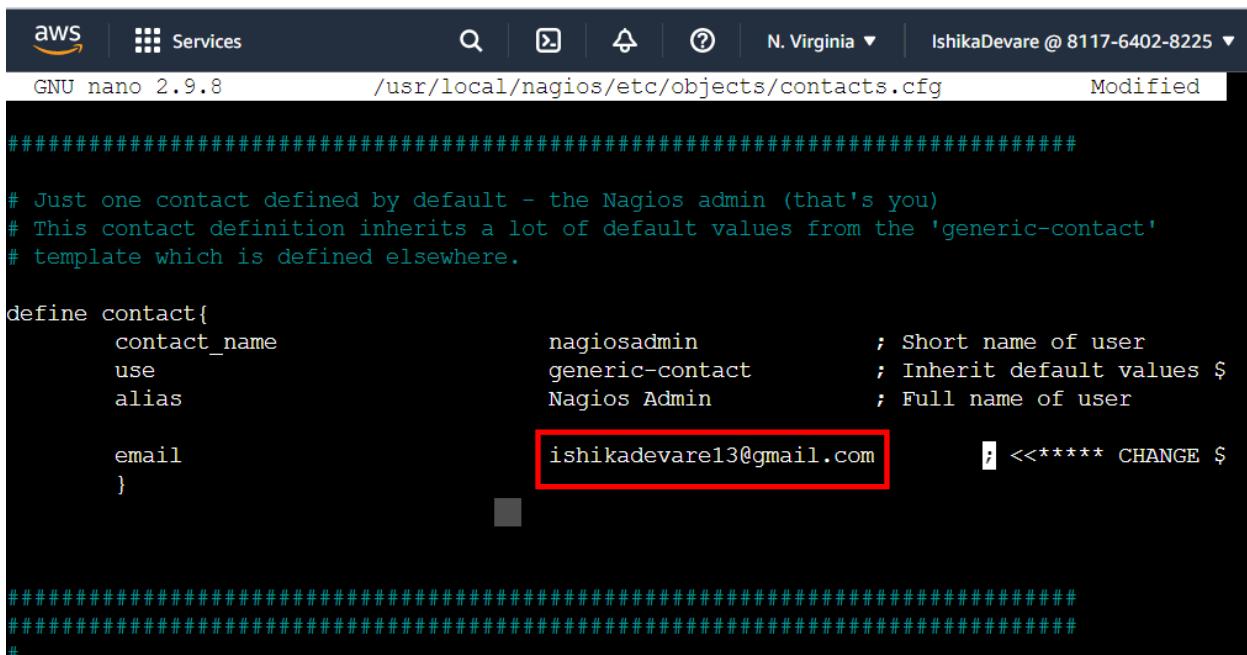
```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagcmd -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

**Step 6:** Customize Configuration Change EMail address from nagiosadmin to your email address.

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



```
aws | Services | Q | S | A | ⓘ | N. Virginia | IshikaDevare @ 8117-6402-8225
GNU nano 2.9.8 /usr/local/nagios/etc/objects/contacts.cfg Modified

#####
# 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 $_
    alias                Nagios Admin          ; Full name of user
    email                ishikadevare13@gmail.com ;<<***** CHANGE $_
}

#####
#
```

**Step 7:** Configure the Web Interface

**sudo make install-webconf**

```
[ec2-user@ip-172-31-81-219 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-172-31-81-219 nagios-4.0.8]$
```

Create a nagiosadmin account for logging into the Nagios web interface. Note the password you need while login to Nagios web console.

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

```
[ec2-user@ip-172-31-81-219 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-172-31-81-219 nagios-4.0.8]$
```

Now Restart the Apache  
**sudo service httpd restart**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

**Step 8:** Compile and Install the Nagios Plugins  
Extract the Nagios plugins source code tarball.

**cd ~/downloads**  
**tar zxvf nagios-plugins-2.0.3.tar.gz**

```
[ec2-user@ip-172-31-81-219 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
```

**cd nagios-plugins-2.0.3**

```
[ec2-user@ip-172-31-81-219 downloads]$ ls
nagios-4.0.5.tar.gz  nagios-4.0.8.tar.gz  nagios-plugins-2.0.3.tar.gz
nagios-4.0.8          nagios-plugins-2.0.3
[ec2-user@ip-172-31-81-219 downloads]$
```

Compile and install the plugins.

## **./configure --with-nagios-user=nagios --with-nagios-group=nagios**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ ./configure --with-nagios-user=nagios --with-nagios-group=nagios
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
checking whether time.h and sys/time.h may both be included... yes
```

## **make**

```
statusdata-cgi.o comments-cgi.o downtime-cgi.o ../lib/libnagios.a -lgd -lpng -ljpeg -lz -lm
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCGI -o histogram.cgi histogram.c ../common/shared.o getcgi.o cgiutils.o cgiauth.o macros-cgi.o objects-cgi.o xobjects-cgi.o statusdata-cgi.o xstatusdata-cgi.o comments-cgi.o downtime-cgi.o ../lib/libnagios.a -lgd -lpng -ljpeg -lz -lm
make[1]: Leaving directory `/home/ec2-user/downloads/nagios-4.0.8/cgi'
cd ./html && make
make[1]: Entering directory `/home/ec2-user/downloads/nagios-4.0.8/html'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/ec2-user/downloads/nagios-4.0.8/html'
if [ xyes = xyes ]; then \
    cd ./module && make; \
fi
make[1]: Entering directory `/home/ec2-user/downloads/nagios-4.0.8/module'
make[1]: Nothing to be done for `all'.
make[1]: Leaving directory `/home/ec2-user/downloads/nagios-4.0.8/module'
cd ./worker && make all
make[1]: Entering directory `/home/ec2-user/downloads/nagios-4.0.8/worker'
cd ./ping && make all
make[2]: Entering directory `/home/ec2-user/downloads/nagios-4.0.8/worker/ping'
make[2]: Nothing to be done for `all'.
make[2]: Leaving directory `/home/ec2-user/downloads/nagios-4.0.8/worker/ping'
make[1]: Leaving directory `/home/ec2-user/downloads/nagios-4.0.8/worker'
```

## **sudo make install**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo make install
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 nagiostats /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/nagiostats
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'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/sbin
for file in *.cgi; do \
    /usr/bin/install -c -m 775 -o nagios -g nagios $file /usr/local/nagios/sbin; \
done
```

## Step 9: Start Nagios

Add Nagios to the list of system services and have it automatically start when the system boots.

**sudo chkconfig --add nagios**

**sudo chkconfig nagios on**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo chkconfig --add nagios
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo chkconfig nagios on
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ █
```

Verify the sample Nagios configuration files.

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

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.0.8
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 08-12-2014
License: GPL

Website: http://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
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

If there are no errors, start Nagios.

**sudo service nagios start**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ sudo service nagios start
Starting nagios (via systemctl): [ OK ]
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

Check the status of Nagios

**sudo systemctl status nagios**

```
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$ 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 Sat 2022-10-01 10:49:25 UTC; 5min ago
    Docs: man:systemd-sysv-generator(8)
 Process: 16166 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
  CGroup: /system.slice/nagios.service
          └─16187 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              ├─16189 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio...
              ├─16190 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio...
              ├─16191 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio...
              ├─16192 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagio...
              └─16193 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 01 10:53:10 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;S...
Oct 01 10:53:17 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;H...
Oct 01 10:53:32 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;R...
Oct 01 10:53:40 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;C...
Oct 01 10:53:47 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;S...
Oct 01 10:53:55 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;P...
Oct 01 10:54:10 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;S...
Oct 01 10:54:17 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;H...
Oct 01 10:54:25 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;T...
Oct 01 10:54:32 ip-172-31-81-219.ec2.internal nagios[16187]: SERVICE ALERT: localhost;R...
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-81-219 nagios-4.0.8]$
```

**Step 10:** Go back to the EC2 console and copy the Public IP address of your instance.

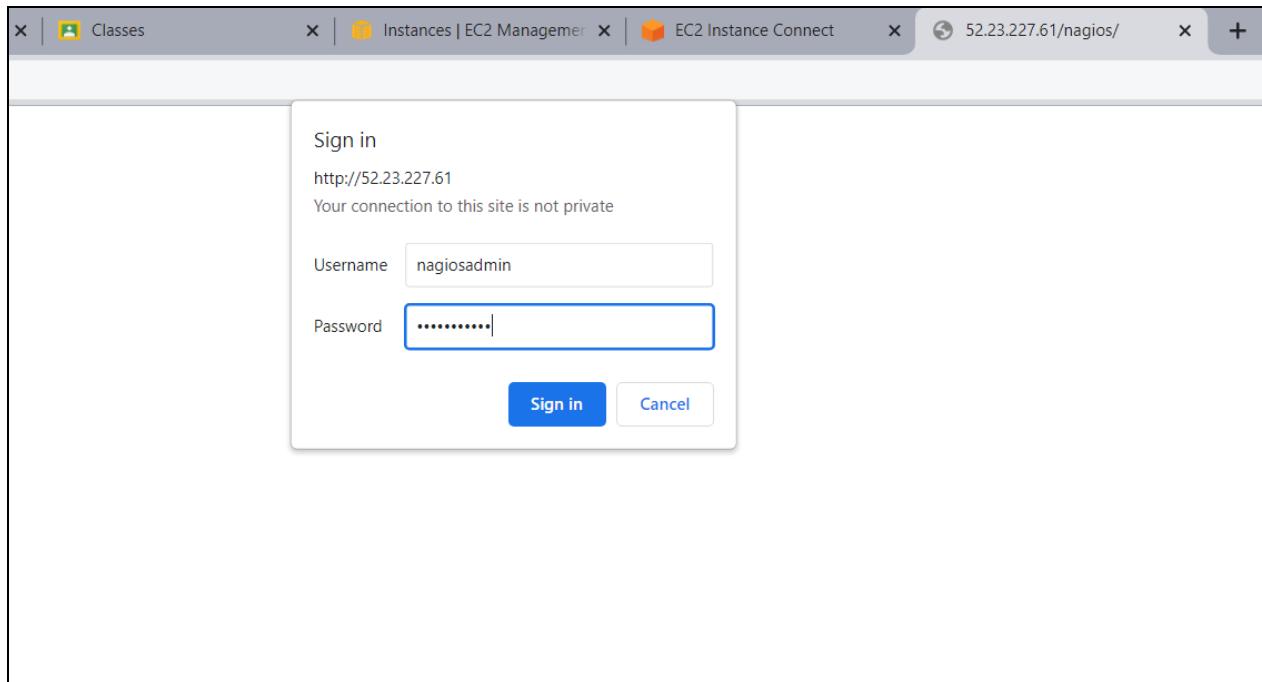
The screenshot shows the AWS EC2 Instances page. A table lists four instances. The second instance, 'AdvDevOps-EXP...', is selected and highlighted with a blue border. The table columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv4 IP. Below the table, a detailed view for the selected instance (i-0dd8cd52d29bfed52) is displayed. The 'Details' tab is active, showing the following information:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0dd8cd52d29bfed52 (AdvDevops-EXP9)	52.23.227.61   open address	172.31.81.219
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-52-23-227-61.compute-1.amazonaws.com   open address
Hostname type	Private IP DNS name (IPv4 only)	
IP name: ip-172-31-81-219.ec2.internal	ip-172-31-81-219.ec2.internal	
Answer private resource DNS name	Instance type	Elastic IP addresses
IPv4 (A)	t2.micro	-

Step 11: Open your browser and look for [http://\\*your public ip address\\*/nagios](http://your public ip address/nagios)  
<http://52.23.227.61/nagios/> (In my case)

The screenshot shows a browser window with multiple tabs open. The active tab is '52.23.227.61/nagios/'. A sign-in dialog box is displayed in the center of the page. The dialog title is 'Sign in' and the URL is 'http://52.23.227.61'. It states 'Your connection to this site is not private'. There are two input fields: 'Username' and 'Password', both currently empty. Below the fields are two buttons: 'Sign In' (highlighted in blue) and 'Cancel'.

Enter username as “nagiosadmin” and password which you set in Step 7.



If all the things are perfect including the credentials , plugin installation then you will be able to see this page

\* Conclusion - In this we got to know about network monitoring, Nagios its plugins etc. We monitored different services of HTTP, SSH etc. Here we have set up nagios software on a AWS Linux machine.