

Screenshots for Experiment 9

Launch an ec2 instance

Give name use the default OS

EC2 > Instances > Launch an instance

Launch an instance

Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Info

Name

nagios_host_exp_9kcs

Add additional tags

▼ Application and OS Images (Amazon Machine Image)

Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

S

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Hitesh Rohra 47 D15A
Make a key pair and use it.

▼ 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_exp_9



[Create new key pair](#)

▼ Network settings [Info](#)

Edit

Network [Info](#)

vpc-07b6966cbfba88ee3

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

We'll create a new security group called '**launch-wizard-5**' with the following rules:

☒ Allow SSH traffic from
Helps you connect to your instance

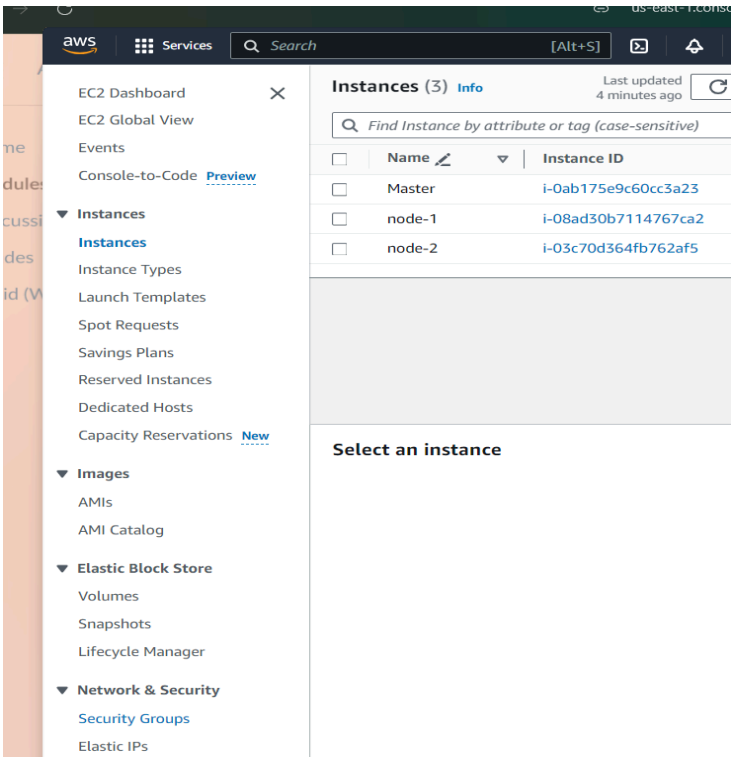
Anywhere
0.0.0.0/0

☐ Allow HTTPS traffic from the internet

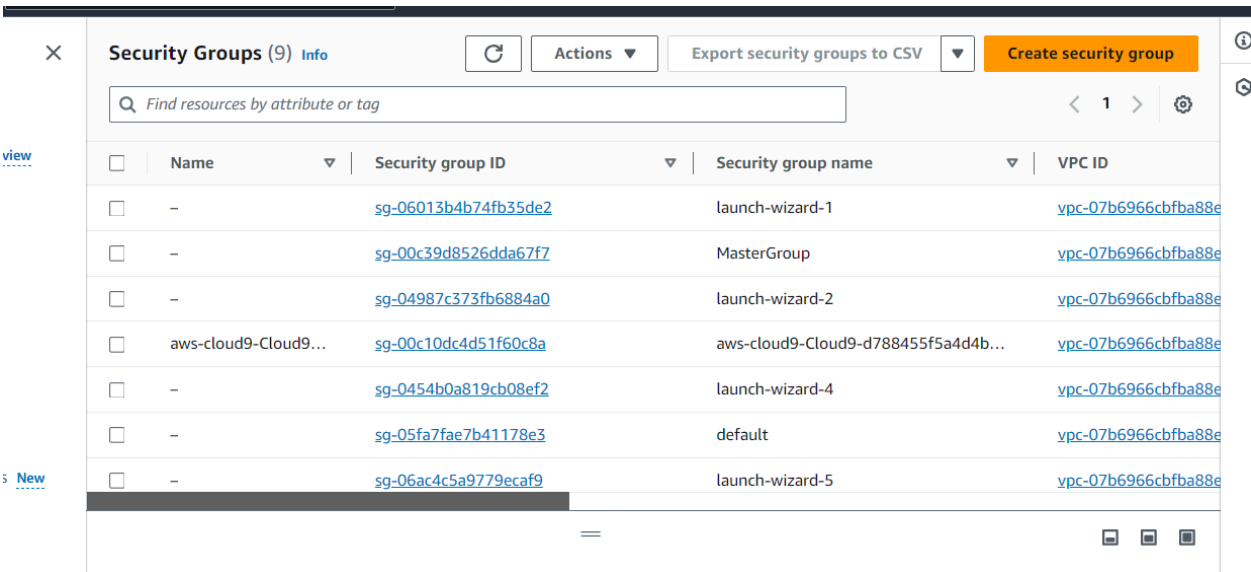
Note the name of the security group that was created for future use:
here it is ' launch-wizard-5 '



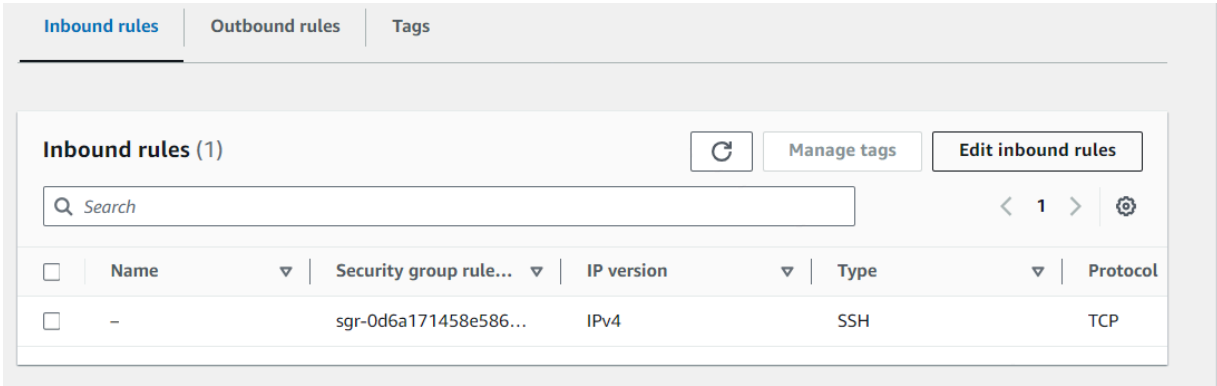
go to security groups:



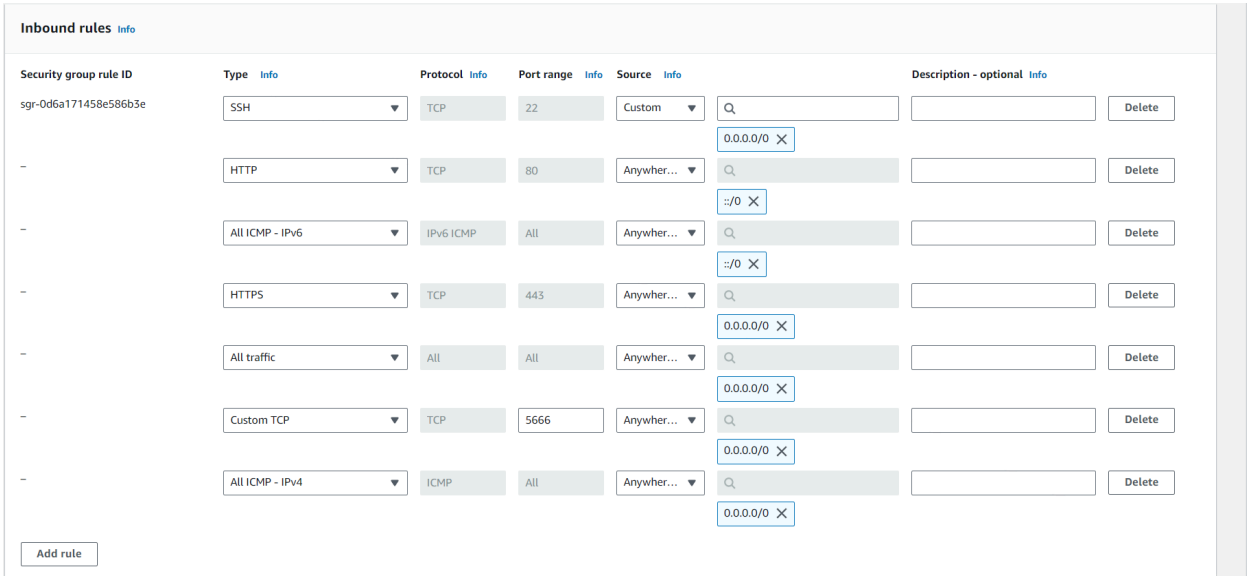
click the security group id which was created while you created the ec2 instance of this experiment.



now click on edit inbound rules



now do the following configurations:
by clicking “add rules”



then click on save rules.

EC2 > Security Groups > sg-06ac4c5a9779ecaf9 - launch-wizard-5

sg-06ac4c5a9779ecaf9 - launch-wizard-5

Details

Security group name
launch-wizard-5

Owner
209322483715

Security group ID
sg-06ac4c5a9779ecaf9

Inbound rules count
7 Permission entries

Description
launch-wizard-5 created 2024-09-28T03:55:31.506Z

Outbound rules count
1 Permission entry

VPC ID
vpc-07b0966c8fba88e3

Inbound rules

Outbound rules

Tags

Inbound rules (7)

Search

Manage tags

Edit inbound rules

	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sg-0576798f8a0b606b	IPv6	All ICMP - IPv6	IPv6 ICMP	All	::/0	-
<input type="checkbox"/>	-	sg-0d6a171458e586...	IPv4	SSH	TCP	22	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-0b17ca9e096a3b5...	IPv4	HTTP5	TCP	443	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-0d5d5a2940a2ba0f0	IPv4	Custom TCP	TCP	5656	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-0e782a66d47b344f5	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-00b8da767cde375...	IPv6	HTTP	TCP	80	::/0	-
<input type="checkbox"/>	-	sg-0c81dc37a4a6020e	IPv4	All traffic	All	All	0.0.0.0/0	-

now navigate to instances, click on the instance which was created earlier and click on connect.

Instances (1/4) info

Last updated less than a 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
<input type="checkbox"/>	Master	i-0ab175e9c60cc3a23	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-165-203-193.co...	54.165.203.193	-	-
<input type="checkbox"/>	node-1	i-08ad30b7114767ca2	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-52-23-200-179.co...	52.23.200.179	-	-
<input type="checkbox"/>	node-2	i-03c70d364fb762af5	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-85-164-72.comp...	3.85.164.72	-	-
<input checked="" type="checkbox"/>	nagios_host_e...	i-0820376be204a7fcb	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-54-205-31-174.co...	54.205.31.174	-	-

now copy the ssh command and just replace the .pem file with its actual location in your computer.

EC2 > Instances > i-0820376be204a7fcb > Connect to instance

Connect to instance info

Connect to your instance i-0820376be204a7fcb (nagios_host_exp_9kcs) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-0820376be204a7fcb (nagios_host_exp_9kcs)

1. Open an SSH client.

2. Locate your private key file. The key used to launch this instance is nagios_exp_9.pem

3. Run this command, if necessary, to ensure your key is not publicly viewable.
chmod 400 "nagios_exp_9.pem"

4. Connect to your instance using its Public DNS:
ec2-54-205-31-174.compute-1.amazonaws.com

Example:
ssh -i "nagios_exp_9.pem" ec2-user@ec2-54-205-31-174.compute-1.amazonaws.com

Note:

In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

paste the command in your terminal and enter after replacing the .pem file with its actual location in your system.

```
C:\Users\Lenovo>ssh -i "C:\Users\Lenovo\Downloads\nagios_exp_9.pem" ec2-user@ec2-54-205-31-174.compute-1.amazonaws.com
The authenticity of host 'ec2-54-205-31-174.compute-1.amazonaws.com (54.205.31.174)' can't be established.
ED25519 key fingerprint is SHA256:+oIS6lcV6qE12x8gFgYVvMsB+yc9vN7UEpF6oBt0jw0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-205-31-174.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-172-31-80-137 ~]$ |
```

now paste the following commands in your connected terminal:

```
sudo yum update
```

```

_/_/_'
[ec2-user@ip-172-31-80-137 ~]$ sudo yum update
Last metadata expiration check: 2:21:45 ago on Sat Sep 28 03:59:04 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-80-137 ~]$ |

```

```
sudo yum install httpd php
```

```

ec2-user@ip-172-31-80-137 ~]$ sudo yum install httpd php
Last metadata expiration check: 2:22:53 ago on Sat Sep 28 03:59:04 2024.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository
=====
Installing:
httpd                                  x86_64            2.4.62-1.amzn2023 amazonlinux
php8.3                                x86_64            8.3.10-1.amzn2023.0.1 amazonlinux
Installing dependencies:
apr                                    x86_64            1.7.2-2.amzn2023.0.2 amazonlinux
apr-util                              x86_64            1.6.3-1.amzn2023.0.1 amazonlinux
generic-logos-httpd                  noarch            18.0.0-12.amzn2023.0.3 amazonlinux
httpd-core                            x86_64            2.4.62-1.amzn2023 amazonlinux
httpdfilesystem                       noarch            2.4.62-1.amzn2023 amazonlinux

```

(type y when prompted)

```
sudo yum install gcc glibc glibc-common
```

```
Dependencies resolved.
=====
Package Architecture Version Repository
=====
Installing:
gcc x86_64 11.4.1-2.amzn2023.0.2 amazonlinux
Installing dependencies:
annobin-docs noarch 10.93-1.amzn2023.0.1 amazonlinux
annobin-plugin-gcc x86_64 10.93-1.amzn2023.0.1 amazonlinux
cpp x86_64 11.4.1-2.amzn2023.0.2 amazonlinux
gc x86_64 8.0.4-5.amzn2023.0.2 amazonlinux
glibc-devel x86_64 2.34-52.amzn2023.0.11 amazonlinux
glibc-headers-x86 noarch 2.34-52.amzn2023.0.11 amazonlinux
guile22 x86_64 2.2.7-2.amzn2023.0.3 amazonlinux
kernel-headers x86_64 6.1.109-118.189.amzn2023 amazonlinux
libmpc x86_64 1.2.1-2.amzn2023.0.2 amazonlinux
libtool-ltdl x86_64 2.4.7-1.amzn2023.0.3 amazonlinux
libxcrypt-devel x86_64 4.4.33-7.amzn2023 amazonlinux
make x86_64 1:4.3-5.amzn2023.0.2 amazonlinux

Transaction Summary
=====
Install 13 Packages

Total download size: 52 M
Installed size: 168 M
Is this ok [y/N]: y|
```

sudo yum install gd gd-devel

```
google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
graphite2-devel-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-devel-7.0.0-2.amzn2023.0.1.x86_64
harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64
jbigkit-libs-2.1-21.amzn2023.0.2.x86_64
libICE-1.0.10-6.amzn2023.0.2.x86_64
libX11-1.7.2-3.amzn2023.0.4.x86_64
libX11-devel-1.7.2-3.amzn2023.0.4.x86_64
libXau-1.0.9-6.amzn2023.0.2.x86_64
libXext-1.3.4-6.amzn2023.0.2.x86_64
libXpm-devel-3.5.15-2.amzn2023.0.3.x86_64
libXt-1.2.0-4.amzn2023.0.2.x86_64
libffi-devel-3.4.4-1.amzn2023.0.1.x86_64
libicu-devel-67.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-devel-2.1.4-2.amzn2023.0.5.x86_64
libpng-2:1.6.37-10.amzn2023.0.6.x86_64
libselinux-devel-3.4-5.amzn2023.0.2.x86_64
libtiff-4.4.0-4.amzn2023.0.18.x86_64
libwebp-1.2.4-1.amzn2023.0.6.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
libxml2-devel-2.10.4-1.amzn2023.0.6.x86_64
pcre2-utf16-10.40-1.amzn2023.0.3.x86_64
pcre2-utf32-10.40-1.amzn2023.0.3.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64
xml-common-0.6.3-56.amzn2023.0.2.noarch
xz-devel-5.2.5-9.amzn2023.0.2.x86_64
graphite2-1.3.14-7.amzn2023.0.2.x86_64
harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
harfbuzz-icu-7.0.0-2.amzn2023.0.1.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
libSM-1.2.3-8.amzn2023.0.2.x86_64
libX11-common-1.7.2-3.amzn2023.0.4.noarch
libX11-xcb-1.7.2-3.amzn2023.0.4.x86_64
libXau-devel-1.0.9-6.amzn2023.0.2.x86_64
libXpm-3.5.15-2.amzn2023.0.3.x86_64
libXrender-0.9.10-14.amzn2023.0.2.x86_64
libblkid-devel-2.37.4-1.amzn2023.0.4.x86_64
libicu-67.1-7.amzn2023.0.3.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libmount-devel-2.37.4-1.amzn2023.0.4.x86_64
libpng-devel-2:1.6.37-10.amzn2023.0.6.x86_64
libsepol-devel-3.4-3.amzn2023.0.3.x86_64
libtiff-devel-4.4.0-4.amzn2023.0.18.x86_64
libwebp-devel-1.2.4-1.amzn2023.0.6.x86_64
libxcb-devel-1.13.1-7.amzn2023.0.2.x86_64
pcre2-devel-10.40-1.amzn2023.0.3.x86_64
pcre2-utf32-10.40-1.amzn2023.0.3.x86_64
sysprof-capture-devel-3.40.1-2.amzn2023.0.2.x86_64
xorg-x11-proto-devel-2021.4-1.amzn2023.0.2.noarch
zlib-devel-1.2.11-33.amzn2023.0.5.x86_64

complete!
ec2-user@ip-172-31-80-137 ~$ |
```

sudo adduser -m nagios
sudo passwd nagios

```
Complete!
[ec2-user@ip-172-31-80-137 ~]$ 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-172-31-80-137 ~]$ |
```

(add a password here)
sudo groupadd nagcmd

```
[ec2-user@ip-172-31-80-137 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-80-137 ~]$ |
```

sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache

```
[ec2-user@ip-172-31-80-137 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-80-137 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-80-137 ~]$ |
```

mkdir ~/downloads
cd ~/downloads

```
[ec2-user@ip-172-31-80-137 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-80-137 downloads]$ |
```

wget <https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz>

```
cd ~/downloads
[ec2-user@ip-172-31-80-137 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-09-28 06:27:51-- https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.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: 2065473 (2.0M) [application/x-gzip]
Saving to: 'nagios-4.5.5.tar.gz'

nagios-4.5.5.tar.gz      100%[=====] 1.97M  5.30MB/s  in 0.4s

2024-09-28 06:27:52 (5.30 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/2065473]
[ec2-user@ip-172-31-80-137 downloads]$ |
```


Hitesh Rohra 47 D15A

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

```
[ec2-user@ip-172-31-80-137 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-09-28 06:28:14-- 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%[=====] 2.62M 5.90MB/s in 0.4s

2024-09-28 06:28:15 (5.90 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]

[ec2-user@ip-172-31-80-137 downloads]$ |
```

tar zxvf nagios-4.5.5.tar.gz

```
[ec2-user@ip-172-31-80-137 downloads]$ tar zxvf nagios-4.5.5.tar.gz
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
```

```
nagiosexp9 × + v
[ec2-user@ip-172-31-80-137 downloads]$ ls
nagios-4.5.5 nagios-4.5.5.tar.gz nagios-plugins-2.4.11.tar.gz
[ec2-user@ip-172-31-80-137 downloads]$ |
```

cd nagios-4.5.5

```
nagios-4.5.5 nagios-4.5.5.tar.gz nagios-plugins-2.4.11.
[ec2-user@ip-172-31-80-137 downloads]$ cd nagios-4.5.5
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ |
```

sudo yum install openssl-devel

```
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ sudo yum install openssl-devel
Last metadata expiration check: 2:31:25 ago on Sat Sep 28 03:59:04 2024.
Dependencies resolved.
=====
Package                        Architecture      Version           Repository        Size
=====
Installing:
openssl-devel                  x86_64            1:3.0.8-1.amzn2023.0.14  amazonlinux      3.0 M

Transaction Summary
=====
Install 1 Package

Total download size: 3.0 M
Installed size: 4.7 M
Is this ok [y/N]: y
```

```
Total 18 MB/s | 3.0 MB 00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      : 1/1
  Installing     : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1
  Running scriptlet: openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1
  Verifying      : openssl-devel-1:3.0.8-1.amzn2023.0.14.x86_64 1/1

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

Complete!
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ |
```

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

```

nagiosexp9
[ec2-user@ip-172-31-80-137 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
checking for string.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes

```

make all

```

[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ make all
cd ./base && make
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/base'
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o ./nagios.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmodes.o nebmodes.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -I. -I../lib -I../include -I../include -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
In function 'get_wproc_list'

```

- 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.

sudo make install

Hitesh Rohra 47 D15A
sudo make install-init
sudo make install-config
sudo make install-commandmode

```
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ 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-4.5.5/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 \
```

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

```
nagiosexp9
GNU nano 5.8 /usr/local/nagios/etc/objects/contacts.cfg
#####
# CONTACTS.CFG - SAMPLE CONTACT/CONTACTGROUP DEFINITIONS
#
#
# NOTES: This config file provides you with some example contact and contact
# group definitions that you can reference in host and service
# definitions.
#
# You don't need to keep these definitions in a separate file from your
# other object definitions. This has been done just to make things
# easier to understand.
#
#####
#
#####
# CONTACTS
#
#####
# Just one contact defined by default - the Nagios admin (that's you)
# [ Read 51 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^Y Exit      ^D Read File  ^_ Replace    ^U Paste      ^J Justify    ^_ Go To Line M-E Redo
```

navigate down to email: and change it to your email address.

```
# 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            nagios@localhost ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>

}

#####
```

Hitesh Rohra 47 D15A

```
# 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           2022.shubham.jha@ves.ac.in| ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>
}

#####
#
# CONTACT GROUPS
#####

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo    M-A Set M...
```

press Ctrl+O and then enter.

then press Ctrl +X

```
chmod g+s /usr/local/nagios/var/rw

*** External command directory configured ***

[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ |
```

sudo make install-webconf

```
[ec2-user@ip-172-31-80-137 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-172-31-80-137 nagios-4.5.5]$ |
```

Adding password for nagios admin

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

```
[ec2-user@ip-172-31-80-137 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-172-31-80-137 nagios-4.5.5]$ |
```

sudo service httpd restart

```
Adding password for user nagiosadmin
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-80-137 nagios-4.5.5]$ |
```

cd ~/downloads

tar zxvf nagios-plugins-2.4.11.tar.gz

```
[ec2-user@ip-172-31-80-137 downloads]$ cd ~/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/
```

cd nagios-plugins-2.4.11

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

```
nagiosexp9 x + v
[ec2-user@ip-172-31-80-137 downloads]$ cd 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
checking whether gcc understands -c and -o together... yes
checking whether make supports the include directive... yes (GNU style)
```

make

sudo make install

```
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/plugins-root'
aking install in po
ake[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/po'
usr/bin/mkdir -p /usr/local/nagios/share
ninstalling fr.gmo as /usr/local/nagios/share/locale/fr/LC_MESSAGES/nagios-plugins.mo
ninstalling de.gmo as /usr/local/nagios/share/locale/de/LC_MESSAGES/nagios-plugins.mo
f test "nagios-plugins" = "gettext-tools"; then \
/usr/bin/mkdir -p /usr/local/nagios/share/gettext/po; \
for file in Makefile.in.in remove-potcdate.sin Makevars.template; do \
/usr/bin/install -c -o nagios -g nagios -m 644 ./$file \
/usr/local/nagios/share/gettext/po/$file; \
done; \
for file in Makevars; do \
rm -f /usr/local/nagios/share/gettext/po/$file; \
done; \
se \
: ; \
t
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11/po'
ake[1]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[2]: Entering directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[2]: Nothing to be done for 'install-exec-am'.
ake[2]: Nothing to be done for 'install-data-am'.
ake[2]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
ake[1]: Leaving directory '/home/ec2-user/downloads/nagios-plugins-2.4.11'
```

sudo chkconfig --add nagios

sudo chkconfig nagios on

```
[ec2-user@ip-172-31-80-137 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-172-31-80-137 nagios-plugins-2.4.11]$
```

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

```
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-172-31-80-137 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.
```

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

```
Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-80-137 nagios-plugins-2.4.11]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-80-137 nagios-plugins-2.4.11]$ |
```

sudo systemctl status nagios

```
[ec2-user@ip-172-31-80-137 nagios-plugins-2.4.11]$ 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 Sat 2024-09-28 07:40:16 UTC; 35s ago
     Docs: https://www.nagios.org/documentation
   Process: 71009 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 71010 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 71011 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 5.6M
        CPU: 82ms
    CGroup: /system.slice/nagios.service
            └─71011 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
               71012 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               71013 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               71014 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               71015 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               71016 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initial
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: qh: core query handler registered
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: qh: echo service query handler registered
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: qh: help for the query handler registered
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: wproc: Successfully registered manager as @wproc with query handler
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: wproc: Registry request: name=Core Worker 71015;pid=71015
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: wproc: Registry request: name=Core Worker 71014;pid=71014
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: wproc: Registry request: name=Core Worker 71013;pid=71013
Sep 28 07:40:16 ip-172-31-80-137.ec2.internal nagios[71011]: wproc: Registry request: name=Core Worker 71012;pid=71012
Sep 28 07:40:17 ip-172-31-80-137.ec2.internal nagios[71011]: Successfully launched command file worker with pid 71016
lines 1-28/28 (END)
```

The screenshot displays the AWS Management Console interface. On the left, there is a navigation menu with options like 'EC2 Dashboard', 'Instances', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', 'Images', 'AMI Catalog', and 'Elastic Block Store'. The main content area shows the 'Instance summary for i-0820376be204a7fcb (nagios_host_exp_9kcs)'. The instance is in a 'Running' state. Key details include: Instance ID: i-0820376be204a7fcb, IPv6 address: -, Hostname type: ip-172-31-80-137.ec2.internal, Answer private resource DNS name: IP v4 (A), Auto-assigned IP address: 54.224.175.95 (Public IP), IAM Role: -, IMDSv2: Required. A tooltip is visible over the 'Public IPv4 address copied' button. The right sidebar provides additional information: Private IPv4 addresses: 172.31.80.137, Public IPv4 DNS: ec2-54-224-175-95.compute-1.amazonaws.com, Elastic IP addresses: -, AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations, and Auto Scaling Group name: -.

