

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

Steps to install and configure Nagios in EC2 instance

1. Create an Amazon Linux EC-2 instance and select either existing key pair or create new

**Name and tags** [Info](#)

Name

nagios\_exp9

[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

SUSE Linux

SUS

Q

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**


Amazon Linux 2023 AMI


Free tier eligible ▼

ami-0ebfd941bbafe70c6 (64-bit (x86), uefi-preferred) / ami-00e73ddc3a6fc7dfe (64-bit (Arm), uefi)

Virtualization: hvm    ENA enabled: true    Root device type: ebs


Key pair name - *required*



nagios 





 [Create new key pair](#)









---

**Instances (1)** [Info](#)

Find Instance by attribute or tag (case-sensitive)  All states 


Instance state: running  nagios  Clear filters

Last updated less than a minute ago  Connect Instance state  Actions  [Launch instances](#) 



<input type="checkbox"/>	Name 	Instance ID	Instance state 	Instance type 	Status check	Alarm status	Availability Zone 	Public IPv4 DNS	Public IPv4 ... 	Elastic IP
<input type="checkbox"/>	nagios_exp9	i-0e96193fa5d06b81f	<span>Running</span> 	t2.micro	 Initializing <a href="#">View alarms</a> 		us-east-1b	ec2-34-204-99-38.com...	34.204.99.38	-








2. Under Security Group, make sure HTTP, HTTPS, SSH, ICMP are open from everywhere.

Security groups

 sg-0f16da0b472ddb2cd (launch-wizard-nagios)

▼ Inbound rules

Filter rules  1 

Name	Security group rule ID	Port range	Protocol	Source	Security groups
	sgr-08edf97e1ffb2664e	5666	TCP	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 
	sgr-0962ca7b4d9e94799	All	ICMP	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 
	sgr-02f1d412e6a12e57f	22	TCP	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 
	sgr-089cc7f54505a4ec2	443	TCP	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 
	sgr-0ff99301989f47715	All	ICMPV6	::/0	<a href="#">launch-wizard-nagios</a> 
	sgr-0942ed387d54df72b	All	All	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 
	sgr-03e52350df6c673f0	80	TCP	0.0.0.0/0	<a href="#">launch-wizard-nagios</a> 

3. SSH into Your EC2 instance or simply use **EC2 Instance Connect** from the browser.

## Connect to instance [Info](#)

Connect to your instance i-0e96193fa5d06b81f (nagios\_exp9) using any of these options


EC2 Instance Connect



Session Manager

**SSH client**


EC2 serial console

Instance ID

 **i-0e96193fa5d06b81f** (nagios\_exp9)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is nagios.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.  
 `chmod 400 "nagios.pem"`
4. Connect to your instance using its Public DNS:  
 `ec2-34-204-99-38.compute-1.amazonaws.com`

Example:

 `ssh -i "nagios.pem" ec2-user@ec2-34-204-99-38.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.

Cancel

Connection Successfully installed

```
PS C:\Users\Avan\Downloads> ssh -i "nagios.pem" ec2-user@ec2-34-204-99-38.compute-1.amazonaws.com
The authenticity of host 'ec2-34-204-99-38.compute-1.amazonaws.com (34.204.99.38)' can't be established.
ED25519 key fingerprint is SHA256:q0aFzcRmrRB/szzYeoU6dKBPZ9bCik2raxD9HkAKy8.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-34-204-99-38.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

A newer release of "Amazon Linux" is available.
  Version 2023.5.20241001:
Run "/usr/bin/dnf check-release-update" for full release and version update info

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

[ec2-user@ip-172-31-32-235 ~]$
```

4. Now, we need to install necessary packages that are required to run nagios properly

```
sudo yum install httpd php
```

```
[ec2-user@ip-172-31-32-235 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:02:20 ago on Thu Oct 3 06:48:23 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
<b>Installing:</b>				
httpd	x86_64	2.4.62-1.amzn2023	amazonlinux	48 k
php8.3	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	10 k
<b>Installing dependencies:</b>				
apr	x86_64	1.7.2-2.amzn2023.0.2	amazonlinux	129 k
apr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98 k
generic-logos-httpd	noarch	18.0.0-12.amzn2023.0.3	amazonlinux	19 k
httpd-core	x86_64	2.4.62-1.amzn2023	amazonlinux	1.4 M
httpd-filesystem	noarch	2.4.62-1.amzn2023	amazonlinux	14 k
httpd-tools	x86_64	2.4.62-1.amzn2023	amazonlinux	81 k
libbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
libsodium	x86_64	1.0.19-4.amzn2023	amazonlinux	176 k
libxslt	x86_64	1.1.34-5.amzn2023.0.2	amazonlinux	241 k
mailcap	noarch	2.1.49-3.amzn2023.0.3	amazonlinux	33 k
nginx-filesystem	noarch	1:1.24.0-1.amzn2023.0.4	amazonlinux	9.8 k
php8.3-cli	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	3.7 M
php8.3-common	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	737 k
php8.3-process	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	45 k
php8.3-xml	x86_64	8.3.10-1.amzn2023.0.1	amazonlinux	154 k

Available Versions:

Version 2023.5.20241001:

Run the following command to upgrade to 2023.5.20241001:

```
dnf upgrade --releasever=2023.5.20241001
```

Release notes:

<https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.5.20241001.html>

```
Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-2.4.62-1.amzn2023.x86_64
httpd-filesystem-2.4.62-1.amzn2023.noarch
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
libxslt-1.1.34-5.amzn2023.0.2.x86_64
mod_http-2.0.27-1.amzn2023.0.3.x86_64
nginx-filesystem-1:1.24.0-1.amzn2023.0.4.noarch
php8.3-cli-8.3.10-1.amzn2023.0.1.x86_64
php8.3-fpm-8.3.10-1.amzn2023.0.1.x86_64
php8.3-opcache-8.3.10-1.amzn2023.0.1.x86_64
php8.3-process-8.3.10-1.amzn2023.0.1.x86_64
php8.3-xml-8.3.10-1.amzn2023.0.1.x86_64
apr-util-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-core-2.4.62-1.amzn2023.x86_64
httpd-tools-2.4.62-1.amzn2023.x86_64
libsodium-1.0.19-4.amzn2023.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_lua-2.4.62-1.amzn2023.x86_64
php8.3-8.3.10-1.amzn2023.0.1.x86_64
php8.3-common-8.3.10-1.amzn2023.0.1.x86_64
php8.3-mbstring-8.3.10-1.amzn2023.0.1.x86_64
php8.3-pdo-8.3.10-1.amzn2023.0.1.x86_64
php8.3-sodium-8.3.10-1.amzn2023.0.1.x86_64
```

## sudo yum install gcc glibc glibc-common

```
[ec2-user@ip-172-31-32-235 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:03:23 ago on Thu Oct 3 06:48:23 2024.
Package glibc-2.34-52.amzn2023.0.11.x86_64 is already installed.
Package glibc-common-2.34-52.amzn2023.0.11.x86_64 is already installed.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
<b>Installing:</b>				
gcc	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	32 M
<b>Installing dependencies:</b>				
annobin-docs	noarch	10.93-1.amzn2023.0.1	amazonlinux	92 k
annobin-plugin-gcc	x86_64	10.93-1.amzn2023.0.1	amazonlinux	887 k
cpp	x86_64	11.4.1-2.amzn2023.0.2	amazonlinux	10 M
gc	x86_64	8.0.4-5.amzn2023.0.2	amazonlinux	105 k
glibc-devel	x86_64	2.34-52.amzn2023.0.11	amazonlinux	27 k
glibc-headers-x86	noarch	2.34-52.amzn2023.0.11	amazonlinux	427 k
guile22	x86_64	2.2.7-2.amzn2023.0.3	amazonlinux	6.4 M
kernel-headers	x86_64	6.1.109-118.189.amzn2023	amazonlinux	1.4 M
libmpc	x86_64	1.2.1-2.amzn2023.0.2	amazonlinux	62 k
libtool-ltdl	x86_64	2.4.7-1.amzn2023.0.3	amazonlinux	38 k
libxcrypt-devel	x86_64	4.4.33-7.amzn2023	amazonlinux	32 k
make	x86_64	1:4.3-5.amzn2023.0.2	amazonlinux	534 k

```
WARNING:
  A newer release of "Amazon Linux" is available.

Available Versions:

Version 2023.5.20241001:
  Run the following command to upgrade to 2023.5.20241001:

    dnf upgrade --releasever=2023.5.20241001

Release notes:
  https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.5.20241001.html

=====

Installed:
annobin-docs-10.93-1.amzn2023.0.1.noarch
cpp-11.4.1-2.amzn2023.0.2.x86_64
gcc-11.4.1-2.amzn2023.0.2.x86_64
glibc-headers-x86-2.34-52.amzn2023.0.11.noarch
kernel-headers-6.1.109-118.189.amzn2023.x86_64
libtool-ltdl-2.4.7-1.amzn2023.0.3.x86_64
make-1:4.3-5.amzn2023.0.2.x86_64
annobin-plugin-gcc-10.93-1.amzn2023.0.1.x86_64
gc-8.0.4-5.amzn2023.0.2.x86_64
glibc-devel-2.34-52.amzn2023.0.11.x86_64
guile22-2.2.7-2.amzn2023.0.3.x86_64
libmpc-1.2.1-2.amzn2023.0.2.x86_64
libxcrypt-devel-4.4.33-7.amzn2023.x86_64

Complete!
```

```
sudo yum install gd gd-devel
```

```
[ec2-user@ip-172-31-32-235 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:04:23 ago on Thu Oct 3 06:48:23 2024.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository        Size
=====
Installing:
gd                                     x86_64            2.3.3-5.amzn2023.0.3  amazonlinux      139 k
gd-devel                             x86_64            2.3.3-5.amzn2023.0.3  amazonlinux      38 k
Installing dependencies:
brotli                               x86_64            1.0.9-4.amzn2023.0.2  amazonlinux      314 k
brotli-devel                         x86_64            1.0.9-4.amzn2023.0.2  amazonlinux      31 k
bzip2-devel                          x86_64            1.0.8-6.amzn2023.0.2  amazonlinux      214 k
cairo                                x86_64            1.17.6-2.amzn2023.0.1  amazonlinux      684 k
cmake-filesystem                     x86_64            3.22.2-1.amzn2023.0.4  amazonlinux      16 k
fontconfig                           x86_64            2.13.94-2.amzn2023.0.2  amazonlinux      273 k
fontconfig-devel                     x86_64            2.13.94-2.amzn2023.0.2  amazonlinux      128 k
fonts-filesystem                     noarch            1:2.0.5-12.amzn2023.0.2  amazonlinux      9.5 k
freetype                             x86_64            2.13.2-5.amzn2023.0.1  amazonlinux      423 k
freetype-devel                       x86_64            2.13.2-5.amzn2023.0.1  amazonlinux      912 k
glib2-devel                          x86_64            2.74.7-689.amzn2023.0.2  amazonlinux      486 k
google-noto-fonts-common             noarch            20201206-2.amzn2023.0.2  amazonlinux      15 k
google-noto-sans-vf-fonts           noarch            20201206-2.amzn2023.0.2  amazonlinux      492 k
graphite2                            x86_64            1.3.14-7.amzn2023.0.2  amazonlinux      97 k
graphite2-devel                     x86_64            1.3.14-7.amzn2023.0.2  amazonlinux      21 k
harfbuzz                             x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      868 k
harfbuzz-devel                       x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      404 k
harfbuzz-icu                        x86_64            7.0.0-2.amzn2023.0.1  amazonlinux      18 k
jbigkit-libs                         x86_64            2.1-21.amzn2023.0.2  amazonlinux      54 k
=====
```

```
Installed:
  brotli-1.0.9-4.amzn2023.0.2.x86_64
  bzip2-devel-1.0.8-6.amzn2023.0.2.x86_64
  cmake-filesystem-3.22.2-1.amzn2023.0.4.x86_64
  fontconfig-devel-2.13.94-2.amzn2023.0.2.x86_64
  freetype-2.13.2-5.amzn2023.0.1.x86_64
  gd-2.3.3-5.amzn2023.0.3.x86_64
  glib2-devel-2.74.7-689.amzn2023.0.2.x86_64
  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
  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
  brotli-devel-1.0.9-4.amzn2023.0.2.x86_64
  cairo-1.17.6-2.amzn2023.0.1.x86_64
  fontconfig-2.13.94-2.amzn2023.0.2.x86_64
  fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch
  freetype-devel-2.13.2-5.amzn2023.0.1.x86_64
  gd-devel-2.3.3-5.amzn2023.0.3.x86_64
  google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
  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
```

5. Create a new Nagios User with its password. You'll have to enter the password twice for confirmation. This is require as to give separate permissions for nagios

**sudo adduser -m nagios**

**sudo passwd nagios**

```
[ec2-user@ip-172-31-32-235 ~]$ 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:
Sorry, passwords do not match.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

6. Create a new user group and use these commands so that you don't have to use sudo for Apache and Nagios

**sudo groupadd nagcmd**

**sudo usermod -a -G nagcmd nagios**

**sudo usermod -a -G nagcmd apache**

```
[ec2-user@ip-172-31-32-235 ~]$ sudo groupadd nagcmd
[ec2-user@ip-172-31-32-235 ~]$ sudo usermod -a -G nagcmd nagios
sudo usermod -a -G nagcmd apache
[ec2-user@ip-172-31-32-235 ~]$ |
```

7. Create a new directory for Nagios downloads

**mkdir ~/downloads**

**cd ~/downloads**

```
[ec2-user@ip-172-31-32-235 ~]$ mkdir ~/downloads
cd ~/downloads
[ec2-user@ip-172-31-32-235 downloads]$ |
```

8. Use wget to download the installation source zip files.

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

```
[ec2-user@ip-172-31-32-235 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.5.tar.gz
--2024-10-03 06:57:29-- 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:fe7:45ce
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  5.21MB/s  in 0.4s

2024-10-03 06:57:29 (5.21 MB/s) - 'nagios-4.5.5.tar.gz' saved [2065473/2065473]
```

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

```
[ec2-user@ip-172-31-32-235 downloads]$ wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
--2024-10-03 06:57:45-- 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  5.29MB/s  in 0.5s

2024-10-03 06:57:46 (5.29 MB/s) - 'nagios-plugins-2.4.11.tar.gz' saved [2753049/2753049]
```

9. Use tar to unzip the downloaded archive and change to that directory.

tar zxvf nagios-4.5.5.tar.gz

```
[ec2-user@ip-172-31-32-235 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/Legal
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
nagios-4.5.5/autoconf-macros/ax_nagios_get_files
nagios-4.5.5/autoconf-macros/ax_nagios_get_inetd
nagios-4.5.5/autoconf-macros/ax_nagios_get_init
nagios-4.5.5/autoconf-macros/ax_nagios_get_os
```

10. Navigate to the extracted folder and Run the configuration script with the same group name you previously created.

cd nagios-4.5.5 (check your own version using `nagios -v` or `ls`)

```
[ec2-user@ip-172-31-32-235 downloads]$ ls
nagios-4.5.5  nagios-4.5.5.tar.gz  nagios-plugins-2.4.11.tar.gz
[ec2-user@ip-172-31-32-235 downloads]$ |
```

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

```
[ec2-user@ip-172-31-32-235 downloads]$ cd nagios-4.5.5
[ec2-user@ip-172-31-32-235 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 library containing connect... none required
checking for initgroups... yes
checking for setenv... yes
checking for strdup... yes
checking for strstr... yes
checking for strtoul... yes
checking for unsetenv... yes
checking for type of socket size... size_t
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-172-31-32-235 nagios-4.5.5]$ |
```

We got error, because ssl headers library is not installed

It can be installed using `sudo yum install openssl-devel`



```
[ec2-user@ip-172-31-32-235 nagios-4.5.5]$ sudo yum install openssl-devel
Last metadata expiration check: 0:54:32 ago on Thu Oct 3 06:48:23 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
Downloading Packages:
openssl-devel-3.0.8-1.amzn2023.0.14.x86_64.rpm                12 MB/s | 3.0 MB    00:00
-----
Total                                                         10 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
```

Now rerun

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

```
*** Configuration summary for nagios 4.5.5 2024-09-17 ***:

General Options:
-----
Nagios executable: nagios
Nagios user/group: nagios,nagios
Command user/group: nagios,nagcmd
Event Broker: yes
Install ${prefix}: /usr/local/nagios
Install ${includedir}: /usr/local/nagios/include/nagios
Lock file: /run/nagios.lock
Check result directory: /usr/local/nagios/var/spool/checkresults
Init directory: /lib/systemd/system
Apache conf.d directory: /etc/httpd/conf.d
Mail program: /bin/mail
Host OS: linux-gnu
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.
```

11. Compile the source code.

make all

```
*** 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.

/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-172-31-32-235 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 ***
```

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

**sudo make install**

**sudo make install-init**

**sudo make install-config**

**sudo make install-commandmode**

```
[ec2-user@ip-172-31-32-235 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 /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
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.5.5/cgi'
cd ./html && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.5.5/html'
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/media
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/stylesheets
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/contexthelp
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/docs/images
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/js
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/images/logos
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/includes
```

13. Edit the config file and change the email address so that we can receive timely alerts about the status of our system.

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

```
GNU nano 5.8 /usr/local/nagios/etc/objects/contacts.cfg Modified
#
#   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)
# 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.avan.shetty@ves.ac.in ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}
}
```

And change email with your email

14. Configure the web interface. This is used to set up web server configuration of nagios dashboard.

**sudo make install-webconf**

```
[ec2-user@ip-172-31-32-235 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 ***
```

15. Create a nagiosadmin account for nagios login along with password. You'll have to specify the password twice.

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

```
[ec2-user@ip-172-31-32-235 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
```

16. Restart Apache

**sudo service httpd restart**

```
[ec2-user@ip-172-31-32-235 nagios-4.5.5]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-32-235 nagios-4.5.5]$ |
```

17. Go back to the downloads folder and unzip the plugins zip file.

**cd ~/downloads**

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

```
[ec2-user@ip-172-31-32-235 nagios-4.5.5]$ 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/
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/
nagios-plugins-2.4.11/config_test/Makefile
nagios-plugins-2.4.11/config_test/run_tests
nagios-plugins-2.4.11/config_test/child_test.c
nagios-plugins-2.4.11/gl/
nagios-plugins-2.4.11/gl/m4/
nagios-plugins-2.4.11/gl/m4/00gnulib.m4
nagios-plugins-2.4.11/gl/m4/absolute-header.m4
nagios-plugins-2.4.11/gl/m4/alloca.m4
nagios-plugins-2.4.11/gl/m4/arpa_inet_h.m4
```

## 18. Compile and install plugins

**cd nagios-plugins-2.4.11****./configure --with-nagios-user=nagios --with-nagios-group=nagios**

```
[ec2-user@ip-172-31-32-235 downloads]$ cd nagios-plugins-2.4.11
[ec2-user@ip-172-31-32-235 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
```

## 19. Start Nagios and Add Nagios to the list of system services

**sudo chkconfig --add nagios****sudo chkconfig nagios on**

```
[ec2-user@ip-172-31-32-235 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-172-31-32-235 nagios-plugins-2.4.11]$
```

## Verify the sample configuration files

```
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
```

As we can see no errors were detected

**sudo service nagios start**

```
[ec2-user@ip-172-31-32-235 nagios-plugins-2.4.11]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-32-235 nagios-plugins-2.4.11]$ |
```

20. Check the status of Nagios

**sudo systemctl status nagios**

```
[ec2-user@ip-172-31-32-235 nagios-plugins-2.4.11]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-32-235 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 Fri 2024-10-04 03:42:22 UTC; 1min 5s ago
     Docs: https://www.nagios.org/documentation
   Process: 20075 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 20076 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 20077 (nagios)
     Tasks: 6 (limit: 1112)
    Memory: 5.9M
       CPU: 92ms
   CGroup: /system.slice/nagios.service
           └─20077 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─20078 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               └─20079 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                 └─20080 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                   └─20081 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
                     └─20082 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: qh: Socket '/usr/local/nagios/var/rw/nagios.qh' successfully initialized
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: qh: core query handler registered
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: qh: echo service query handler registered
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: qh: help for the query handler registered
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: wproc: Successfully registered manager as @wproc with query handler
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: wproc: Registry request: name=Core Worker 20079;pid=20079
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: wproc: Registry request: name=Core Worker 20080;pid=20080
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: wproc: Registry request: name=Core Worker 20081;pid=20081
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: wproc: Registry request: name=Core Worker 20078;pid=20078
Oct 04 03:42:22 ip-172-31-32-235.ec2.internal nagios[20077]: Successfully launched command file worker with pid 20082
```

The nagios service is running and working normally

21. Open up your browser and look for [http://<your\\_public\\_ip\\_address>/nagios](http://<your_public_ip_address>/nagios)

Enter username as **nagiosadmin** and password which we had set previously

22. After entering the correct credentials, you will see the Home page of Nagios.

**Nagios® Core™**  
✓ Daemon running with PID 20077

**Nagios® Core™**  
**Version 4.5.5**  
September 17, 2024  
[Check for updates](#)

**Get Started**

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

**Quick Links**

- Nagios Library (tutorials and docs)
- Nagios Labs (development blog)
- Nagios Exchange (plugins and addons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

**Latest News**

**Don't Miss...**

Copyright © 2010-2024 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, servicemarks, registered trademarks or registered servicemarks owned by Nagios Enterprises, LLC. Use of the Nagios marks is governed by the trademark use restrictions.

Now we can see system logs using nagios

**Current Network Status**  
Last Updated: Fri Oct 4 14:20:53 UTC 2024  
Updated every 90 seconds  
Nagios® Core™ 4.5.5 - [www.nagios.org](http://www.nagios.org)  
Logged in as nagiosadmin

**Host Status Totals**

Up	Down	Unreachable	Pending
1	0	0	0

**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
6	1	0	1	0

**Service Status Details For All Hosts**

Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	10-04-2024 14:15:58	0d 0h 14m 55s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	10-04-2024 14:16:36	0d 0h 14m 17s	1/4	USERS OK - 1 users currently logged in
	HTTP	WARNING	10-04-2024 14:20:13	0d 0h 10m 40s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 319 bytes in 0.001 second response time
	PING	OK	10-04-2024 14:17:51	0d 0h 13m 2s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
	Root Partition	OK	10-04-2024 14:18:28	0d 0h 12m 25s	1/4	DISK OK - free space: / 6031 MIB (74.31% inode=98%):
	SSH	OK	10-04-2024 14:19:06	0d 0h 11m 47s	1/4	SSH OK - OpenSSH 8.7 (protocol 2.0)
	Swap Usage	CRITICAL	10-04-2024 14:17:43	0d 0h 8m 10s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	10-04-2024 14:20:21	0d 0h 10m 32s	1/4	PROCS OK: 37 processes with STATE = RSDZT

Results 1 - 8 of 8 Matching Services

## Conclusion:

The successful installation and configuration of Nagios Core, Nagios Plugins, and NRPE on the EC2 instance marks a significant step towards establishing comprehensive system monitoring capabilities. The ability to effectively track and analyze system performance, resource utilization, and potential issues is crucial for maintaining operational efficiency and preventing downtime. In the course of experiment various issues regarding the nagios version and the ssl error was found and that was fixed by installing the openssl library. Apart from that we got to learn the basic fundamentals of nagios and how the plugins are configured in the system or the host machine. At the end Nagios server was hosted on the ec2 instance with the public ip.