Advanced DevOps Lab
Experiment 9

Name: Yash Rahate
Class: D15B
Roll No.: 48

Aim:

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

Theory:

What is Nagios?
Nagios is an open-source software for continuous monitoring of systems, networks, and infrastructures. It runs plugins stored on a server that is connected with a host or another server on your network or the Internet. In case of any failure, Nagios alerts about the issues so that the technical team can perform the recovery process immediately.
Nagios is used for continuous monitoring of systems, applications, service and business processes in a DevOps culture.

Why Use Nagios?
Key reasons to use Nagios include:

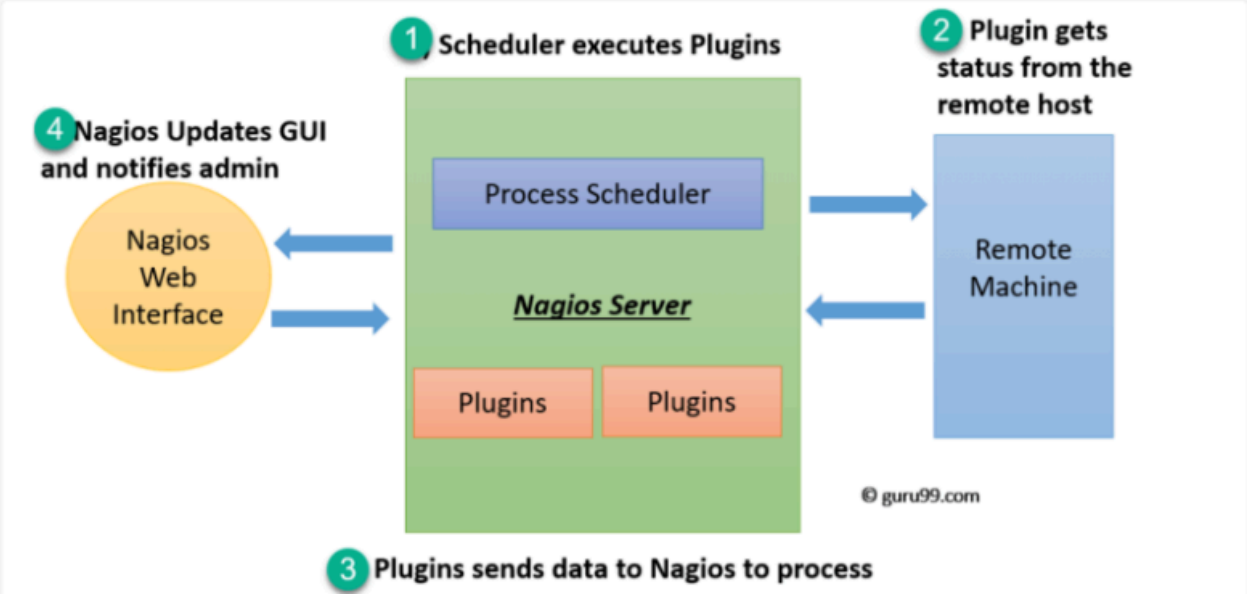
- Detecting network or server issues early
- Identifying the root cause for permanent fixes
- Active monitoring of your infrastructure and processes
- Troubleshooting server performance problems
- Planning upgrades to prevent failures
- Maintaining security and service availability
- Automating problem resolution in critical situations

Features of Nagios:

- Scalable, secure, and manageable
- Attractive web interface
- Automatic alerts for changing conditions
- Ability to monitor network services like HTTP, FTP, SMTP, SSH, and more
- Detecting server crashes and performance issues
- Easy plugin integration
- Monitors entire business processes with a single tool
- Event handlers for proactive issue resolution

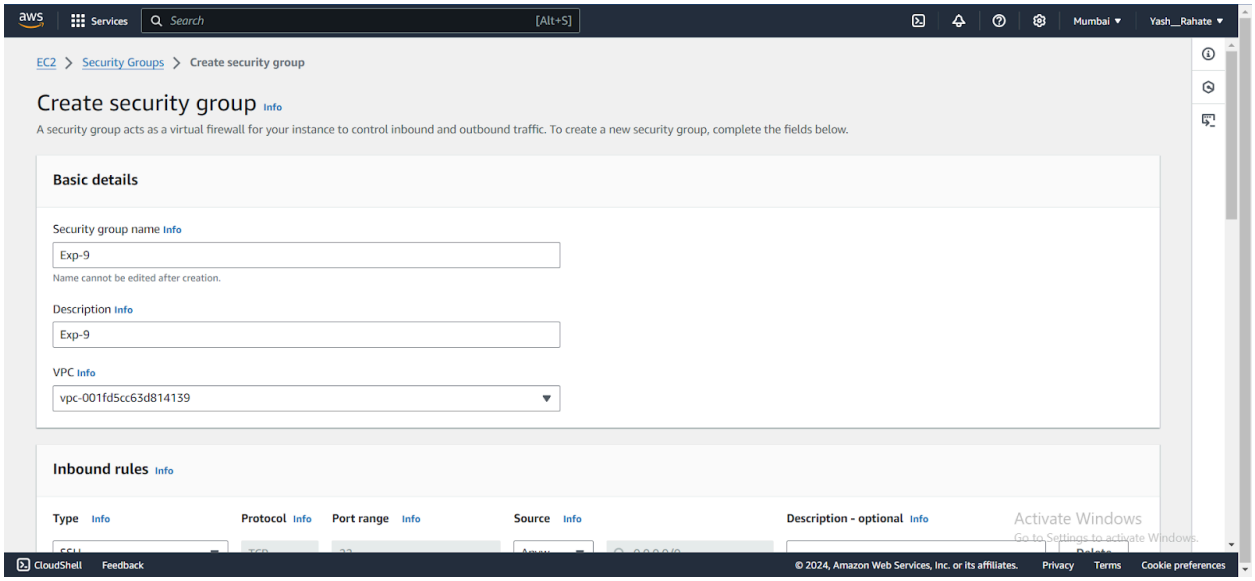
Nagios Architecture

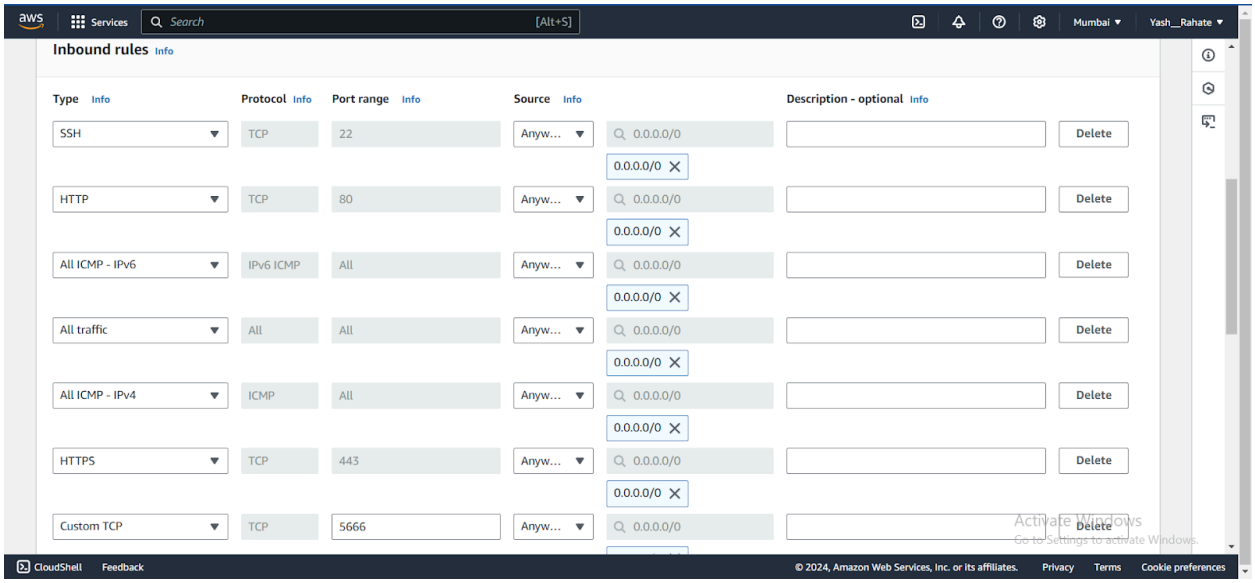
Nagios is a client-server architecture. Usually, on a network, a Nagios server is running on a host, and plugins are running on all the remote hosts which should be monitored.

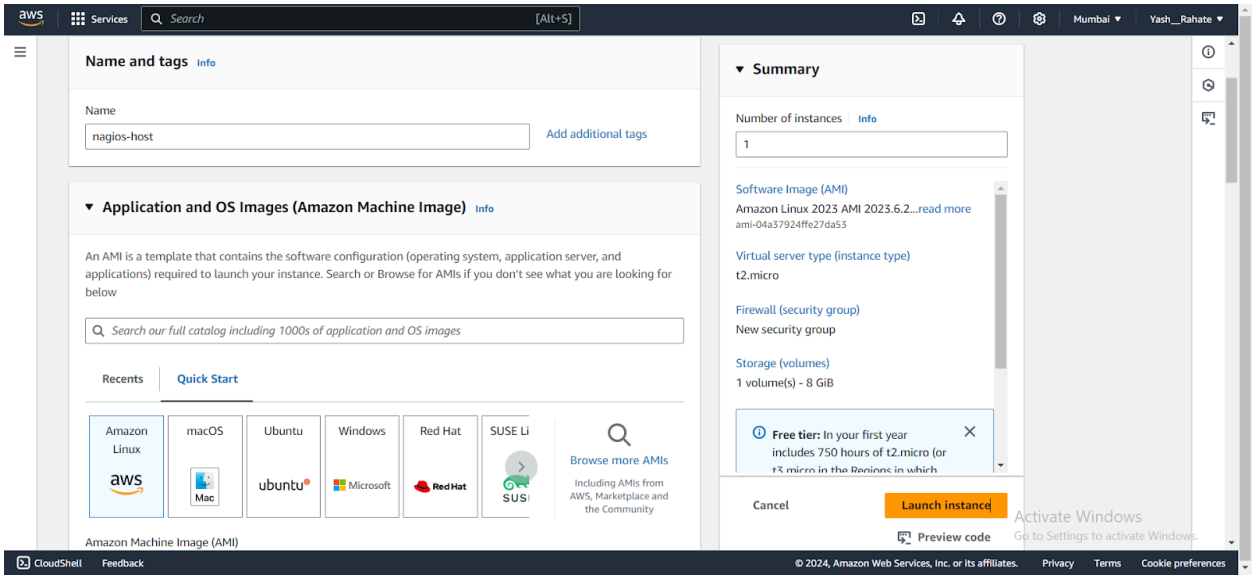


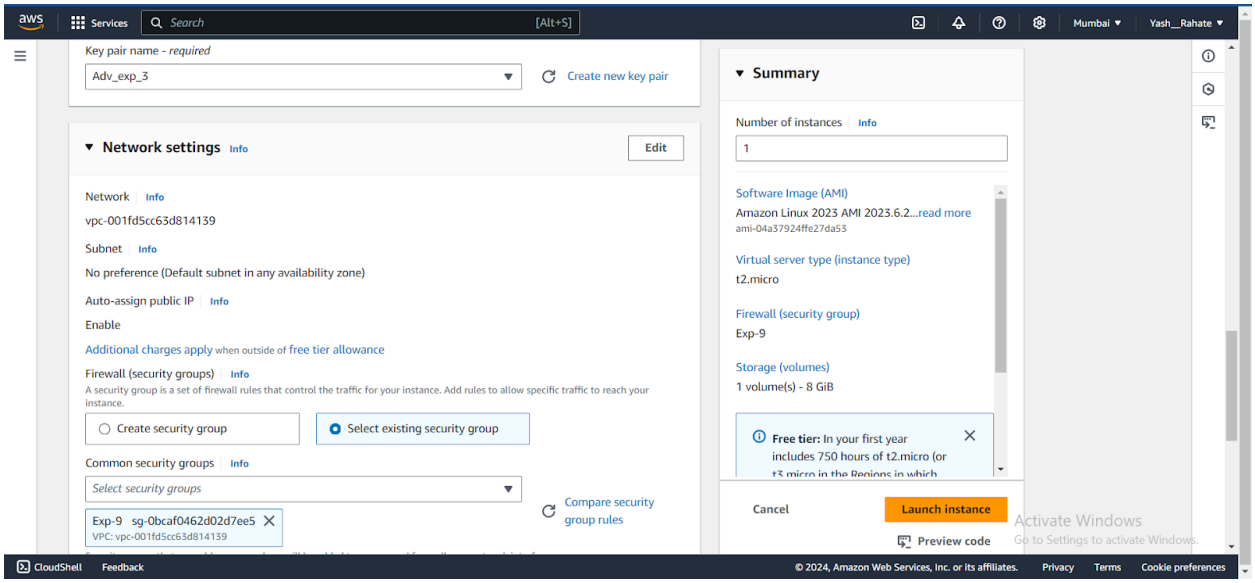
1. The scheduler is a component of the server part of Nagios. It sends a signal to execute the plugins at the remote host.
2. The plugin gets the status from the remote host
3. The plugin sends the data to the process scheduler
4. The process scheduler updates the GUI and notifications are sent to admins.

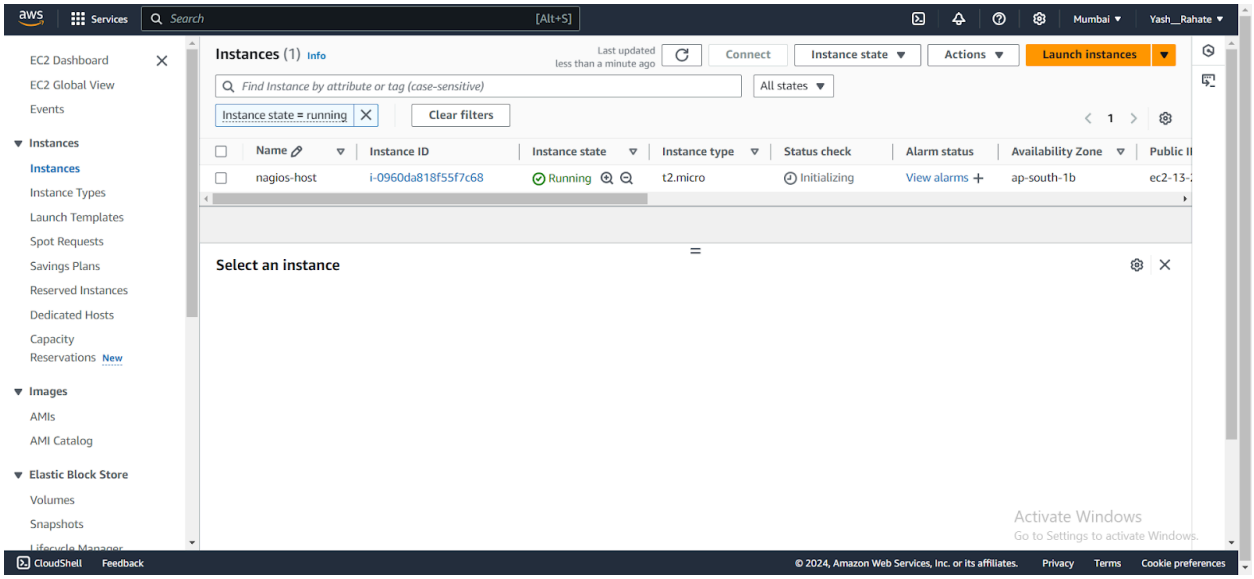
Steps:

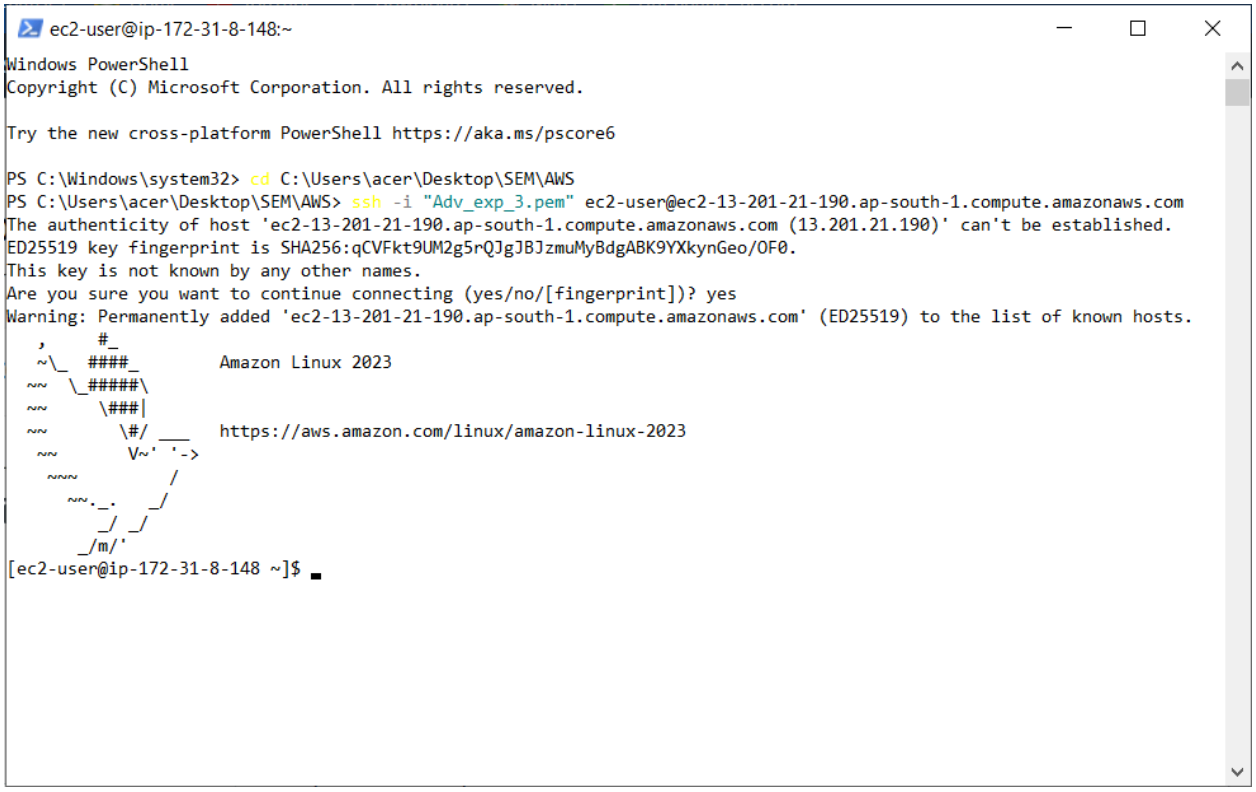

aws
Services
Search
[Alt+S]
Mumbai
Yash__Rahate
EC2 > Security Groups > Create security group
Create security group Info
A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.
Basic details
Security group name Info
Exp-9
Name cannot be edited after creation.
Description Info
Exp-9
VPC Info
vpc-001fd5cc63d814139
Inbound rules Info
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Mumbai
Yash__Rahate
Inbound rules Info
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
SSH
TCP
22
Anyw...
0.0.0.0/0
Delete
HTTP
TCP
80
All ICMP - IPv6
IPv6 ICMP
All
All traffic
All
All
All ICMP - IPv4
ICMP
All
HTTPS
TCP
443
Custom TCP
TCP
5666
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Services
Search
[Alt+S]
Mumbai
Yash_Rahate
Name and tags Info
Name
nagios-host
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
Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-04a37924ffe27da53
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance
Preview code
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Services
Search
[Alt+S]
Mumbai
Yash_Rahate
Key pair name - required
Adv_exp_3
Create new key pair
Network settings Info
Edit
Network Info
vpc-001fd5cc63d814139
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
Common security groups Info
Select security groups
Exp-9 sg-0bcaf0462d02d7ee5
VPC: vpc-001fd5cc63d814139
Compare security group rules
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-04a37924ffe27da53
Virtual server type (instance type)
t2.micro
Firewall (security group)
Exp-9
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or
Cancel
Launch instance
Preview code
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
Mumbai
Yash_Rahate
EC2 Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations New
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Instances (1) Info
Last updated less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
nagios-host
i-0960da818f55f7c68
Running
t2.micro
Initializing
View alarms
ap-south-1b
Select an instance
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


ec2-user@ip-172-31-8-148:~
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Windows\system32> cd C:\Users\acer\Desktop\SEM\AWS
PS C:\Users\acer\Desktop\SEM\AWS> ssh -i "Adv_exp_3.pem" ec2-user@ec2-13-201-21-190.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-13-201-21-190.ap-south-1.compute.amazonaws.com (13.201.21.190)' can't be established.
ED25519 key fingerprint is SHA256:qCVFkt9UM2g5rQJgJBJzmuMyBdgABK9YXkynGeo/OF0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-201-21-190.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-172-31-8-148 ~]$

```
ec2-user@ip-172-31-8-148:~
[ec2-user@ip-172-31-8-148 ~]$ sudo yum install httpd php
Last metadata expiration check: 0:03:28 ago on Thu Oct 17 08:24:13 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture    Version                 Repository    Size
================================================================================
Installing:
 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
Installing dependencies:
 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
Installing weak dependencies:
 apr-util-openssl        x86_64          1.6.3-1.amzn2023.0.1    amazonlinux   17 k
 mod_http2               x86_64          2.0.27-1.amzn2023.0.3   amazonlinux  166 k
 mod_lua                 x86_64          2.4.62-1.amzn2023       amazonlinux   61 k
 php8.3-fpm              x86_64          8.3.10-1.amzn2023.0.1   amazonlinux  1.9 M
 php8.3-mbstring         x86_64          8.3.10-1.amzn2023.0.1   amazonlinux  528 k
 php8.3-opcache          x86_64          8.3.10-1.amzn2023.0.1   amazonlinux  379 k
```

```
ec2-user@ip-172-31-8-148:~
[ec2-user@ip-172-31-8-148 ~]$ sudo yum install gcc glibc glibc-common
Last metadata expiration check: 0:05:23 ago on Thu Oct 17 08:24:13 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
================================================================================
Installing:
 gcc                     x86_64          11.4.1-2.amzn2023.0.2   amazonlinux   32 M
Installing dependencies:
 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          6.1.112-122.189.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

Transaction Summary
================================================================================
Install  13 Packages

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

ec2-user@ip-172-31-8-148:~

```
[ec2-user@ip-172-31-8-148 ~]$ sudo yum install gd gd-devel
Last metadata expiration check: 0:08:48 ago on Thu Oct 17 08:24:13 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
langpacks-core-font-en	noarch	3.0-21.amzn2023.0.4	amazonlinux	10 k
libICE	x86_64	1.1.1-3.amzn2023.0.1	amazonlinux	76 k
libSM	x86_64	1.2.4-3.amzn2023.0.1	amazonlinux	45 k

```
[ec2-user@ip-172-31-8-148 ~]$ sudo adduser -m nagios
[ec2-user@ip-172-31-8-148 ~]$ sudo passwd nagios
Changing password for user nagios.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[ec2-user@ip-172-31-8-148 ~]$
```

```
[ec2-user@ip-172-31-8-148 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.tar.gz
--2024-10-17 08:40:59--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.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: 11339450 (11M) [application/x-gzip]
Saving to: ‘nagios-4.4.9.tar.gz’

nagios-4.4.9.tar.gz          100%[=============================================>]  10.81M  2.36MB/s    in 6.0s

2024-10-17 08:41:06 (1.79 MB/s) - ‘nagios-4.4.9.tar.gz’ saved [11339450/11339450]

[ec2-user@ip-172-31-8-148 downloads]$
```

```
[ec2-user@ip-172-31-8-148 downloads]$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.tar.gz
--2024-10-17 08:40:59--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.9.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: 11339450 (11M) [application/x-gzip]
Saving to: ‘nagios-4.4.9.tar.gz’

nagios-4.4.9.tar.gz          100%[=============================================>]  10.81M  2.36MB/s    in 6.0s

2024-10-17 08:41:06 (1.79 MB/s) - ‘nagios-4.4.9.tar.gz’ saved [11339450/11339450]

[ec2-user@ip-172-31-8-148 downloads]$ wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
--2024-10-17 08:48:36--  http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 45.56.123.251
Connecting to nagios-plugins.org (nagios-plugins.org)|45.56.123.251|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2659772 (2.5M) [application/x-gzip]
Saving to: ‘nagios-plugins-2.0.3.tar.gz’

nagios-plugins-2.0.3.tar.gz  100%[=============================================>]   2.54M   844KB/s    in 3.1s

2024-10-17 08:48:40 (844 KB/s) - ‘nagios-plugins-2.0.3.tar.gz’ saved [2659772/2659772]

[ec2-user@ip-172-31-8-148 downloads]$
```

ec2-user@ip-172-31-8-148:~/downloads

```
[ec2-user@ip-172-31-8-148 downloads]$ tar zxvf nagios-4.4.9.tar.gz
nagios-4.4.9/
nagios-4.4.9/.gitignore
nagios-4.4.9/.travis.yml
nagios-4.4.9/CONTRIBUTING.md
nagios-4.4.9/Changelog
nagios-4.4.9/INSTALLING
nagios-4.4.9/LEGAL
nagios-4.4.9/LICENSE
nagios-4.4.9/Makefile.in
nagios-4.4.9/README.md
nagios-4.4.9/THANKS
nagios-4.4.9/UPGRADING
nagios-4.4.9/aclocal.m4
nagios-4.4.9/autoconf-macros/
nagios-4.4.9/autoconf-macros/.gitignore
nagios-4.4.9/autoconf-macros/CHANGELOG.md
nagios-4.4.9/autoconf-macros/LICENSE
nagios-4.4.9/autoconf-macros/LICENSE.md
nagios-4.4.9/autoconf-macros/README.md
nagios-4.4.9/autoconf-macros/add_group_user
nagios-4.4.9/autoconf-macros/ax_nagios_get_distrib
nagios-4.4.9/autoconf-macros/ax_nagios_get_files
nagios-4.4.9/autoconf-macros/ax_nagios_get_inetd
nagios-4.4.9/autoconf-macros/ax_nagios_get_init
nagios-4.4.9/autoconf-macros/ax_nagios_get_os
nagios-4.4.9/autoconf-macros/ax_nagios_get_paths
nagios-4.4.9/autoconf-macros/ax_nagios_get_ssl
nagios-4.4.9/base/
nagios-4.4.9/base/.gitignore
nagios-4.4.9/base/Makefile.in
```

```
[ec2-user@ip-172-31-8-148 downloads]$ ls
nagios-4.4.9  nagios-4.4.9.tar.gz  nagios-plugins-2.0.3.tar.gz
[ec2-user@ip-172-31-8-148 downloads]$ cd nagios-4.4.9
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ sudo yum install openssl-devel
Last metadata expiration check: 0:28:36 ago on Thu Oct 17 08:24:13 2024.
Dependencies resolved.
================================================================================================================================
 Package                    Architecture        Version                              Repository                 Size
================================================================================================================================
Installing:
 openssl-devel              x86_64              1:3.0.8-1.amzn2023.0.16              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]:
```

ec2-user@ip-172-31-8-148:~/downloads/nagios-4.4.9

```
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ ./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 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 whether ln -s works... 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
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... yes
checking for stdint.h... yes
checking for unistd.h... yes
checking arpa/inet.h usability... yes
```

ec2-user@ip-172-31-8-148:~/downloads/nagios-4.4.9

```
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ make clean
cd ./lib && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/lib'
rm -f untested *.gcov *.gcda *.gcno gmon.out
rm -f test-squeue test-kvvec test-iocache test-iobroker test-bitmap test-dkhash test-runcmd test-nsutils test-fanout
rm -f core.* *.o *~ wproc *.a
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/lib'
cd ./base && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/base'
rm -f nagios nagiostats core *.o gmon.out
rm -f *~ *.*~
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/base'
cd ./cgi && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
rm -f archivejson.cgi avail.cgi cmd.cgi config.cgi extinfo.cgi history.cgi notifications.cgi objectjson.cgi outages.cgi
showlog.cgi status.cgi statusjson.cgi statuswml.cgi summary.cgi tac.cgi statuswrl.cgi statusmap.cgi trends.cgi histogram
.cgi
rm -f *.o core gmon.out
rm -f *~ *.*~
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
cd ./common && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/common'
rm -f core *.o
rm -f *~
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/common'
cd ./xdata && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/xdata'
rm -f *.o
rm -f *~
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/xdata'
cd ./html && make clean
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/html'
```

ec2-user@ip-172-31-8-148:~/downloads/nagios-4.4.9

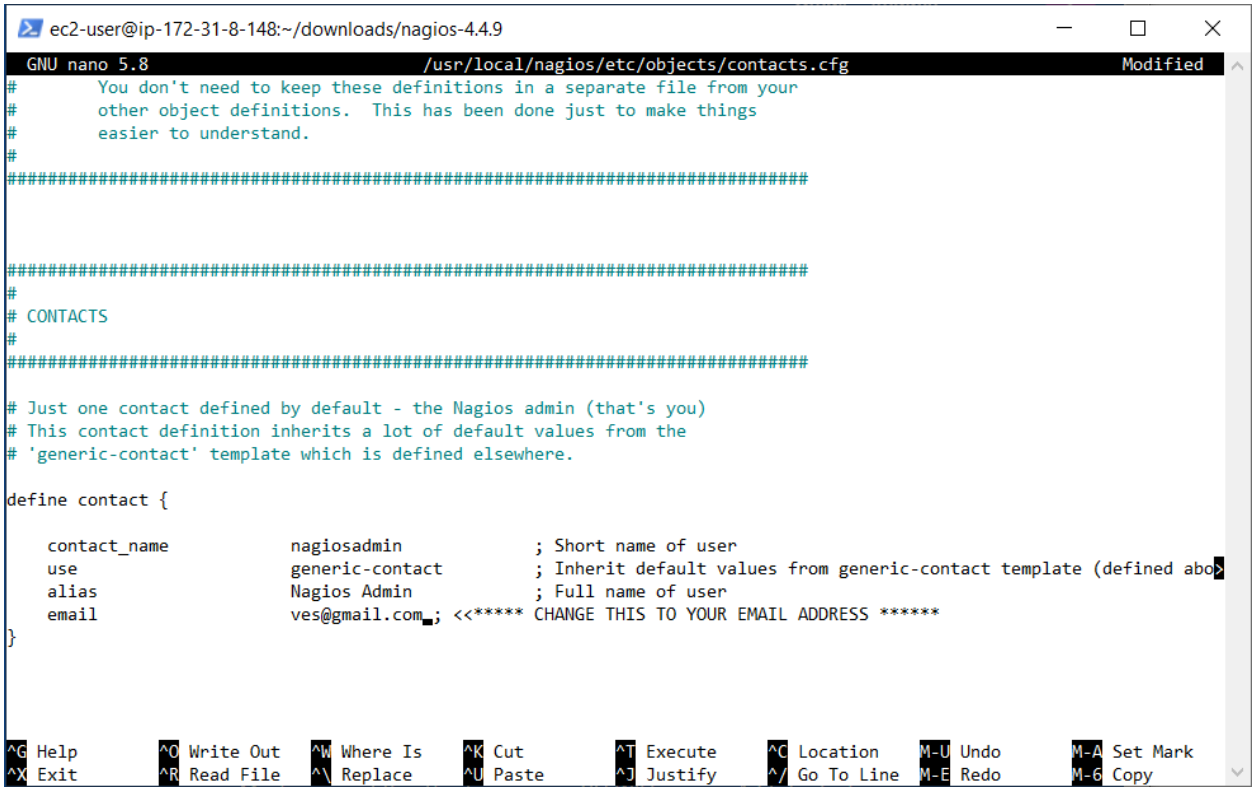
```
rm -f worker-ping worker-ping.o
rm -f core *.o
rm -f *~ *.*~
make[2]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/worker/ping'
rm -f *~ *.*~
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/worker'
rm -f *.cfg core
rm -f *~ *.*~ */*~ */*.*~ */*/*.*~
rm -f nagioscore.info-file
rm -f *.gcno */*.gcno */*/*.gcno
rm -f *.gcda */*.gcda */*/*.gcda
rm -f *.gcov */*.gcov */*/*.gcov
rm -rf coverage-report
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ make all
```

```
ec2-user@ip-172-31-8-148:~/downloads/nagios-4.4.9
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/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 nagiostats /usr/local/nagios/bin
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
make install-basic
make[2]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/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-4.4.9/cgi'
make[1]: Leaving directory '/home/ec2-user/downloads/nagios-4.4.9/cgi'
cd ./html && make install
make[1]: Entering directory '/home/ec2-user/downloads/nagios-4.4.9/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
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/ssi
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/angularjs
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/share/angularjs/angular-1.3.9
```

```
ec2-user@ip-172-31-8-148:~/downloads/nagios-4.4.9
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ 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/obje
cts/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objec
ts/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objec
ts/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/ob
jects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/obje
cts/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/object
s/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/object
s/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects
/switch.cfg

*** Config files installed ***

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

```
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ 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-172-31-8-148 nagios-4.4.9]$
```

```
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ sudo nano /usr/local/nagios/etc/objects/contacts.cfg
```




```
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ 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-172-31-8-148 nagios-4.4.9]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$
```

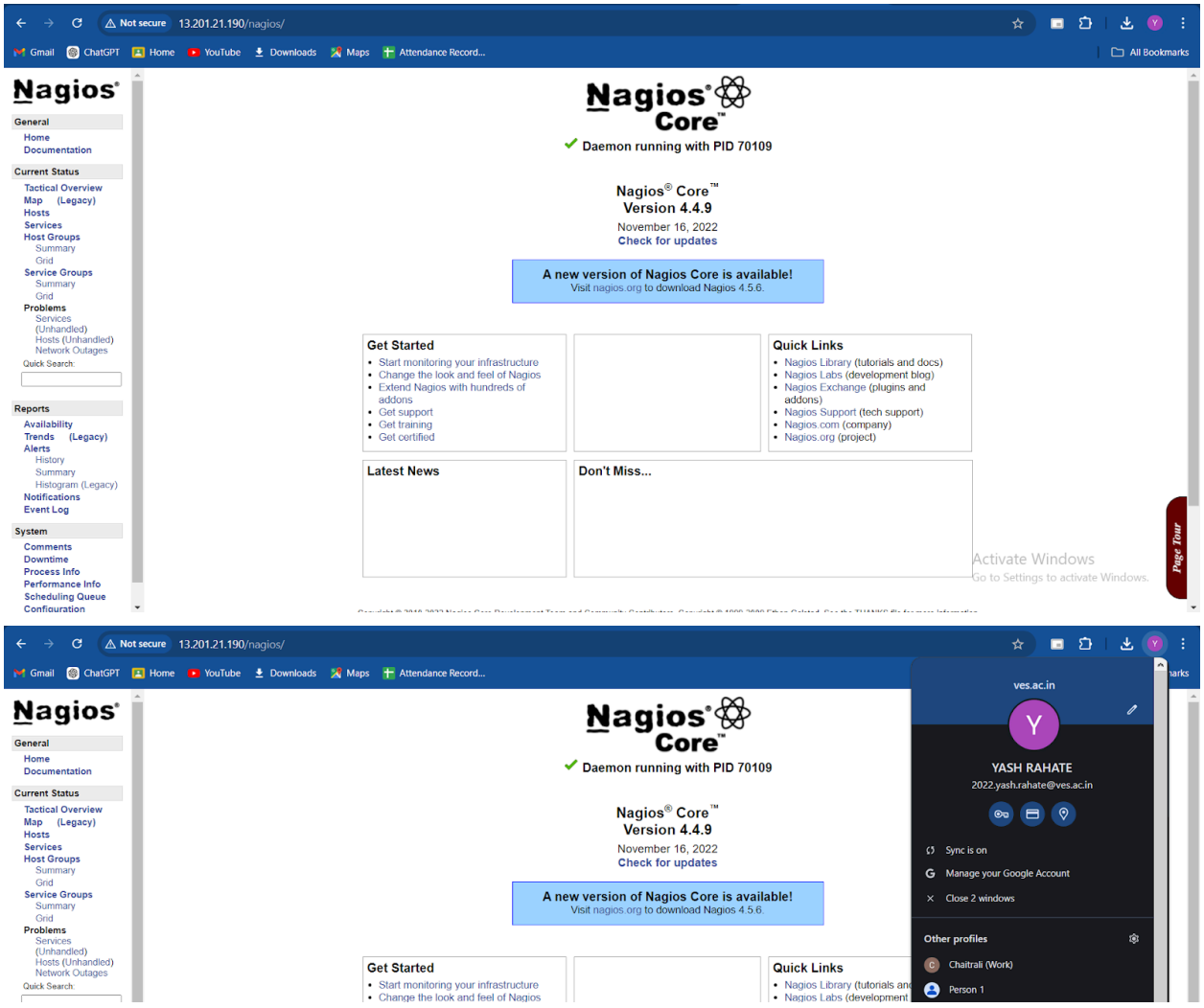
```
ec2-user@ip-172-31-8-148:~/downloads
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[ec2-user@ip-172-31-8-148 nagios-4.4.9]$ cd ~/downloads
[ec2-user@ip-172-31-8-148 downloads]$ tar zxvf nagios-plugins-2.0.3.tar.gz
nagios-plugins-2.0.3/
nagios-plugins-2.0.3/perlmods/
nagios-plugins-2.0.3/perlmods/Config-Tiny-2.14.tar.gz
nagios-plugins-2.0.3/perlmods/parent-0.226.tar.gz
nagios-plugins-2.0.3/perlmods/Test-Simple-0.98.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.in
nagios-plugins-2.0.3/perlmods/version-0.9903.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile.am
nagios-plugins-2.0.3/perlmods/Module-Runtime-0.013.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Metadata-1.000014.tar.gz
nagios-plugins-2.0.3/perlmods/Params-Validate-1.08.tar.gz
nagios-plugins-2.0.3/perlmods/Class-Accessor-0.34.tar.gz
nagios-plugins-2.0.3/perlmods/Try-Tiny-0.18.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Implementation-0.07.tar.gz
nagios-plugins-2.0.3/perlmods/Makefile
nagios-plugins-2.0.3/perlmods/Perl-OSType-1.003.tar.gz
nagios-plugins-2.0.3/perlmods/install_order
nagios-plugins-2.0.3/perlmods/Nagios-Plugin-0.36.tar.gz
nagios-plugins-2.0.3/perlmods/Math-Calc-Units-1.07.tar.gz
nagios-plugins-2.0.3/perlmods/Module-Build-0.4007.tar.gz
nagios-plugins-2.0.3/ABOUT-NLS
nagios-plugins-2.0.3/configure.ac
nagios-plugins-2.0.3/Makefile.in
nagios-plugins-2.0.3/config.h.in
nagios-plugins-2.0.3/ChangeLog
nagios-plugins-2.0.3/AUTHORS
nagios-plugins-2.0.3/lib/
nagios-plugins-2.0.3/lib/parse_ini.h
```

```
[ec2-user@ip-172-31-8-148 nagios-plugins-2.0.3]$ make

[ec2-user@ip-172-31-8-148 nagios-plugins-2.0.3]$ sudo make install
```

```
[ec2-user@ip-172-31-8-148 ~]$ sudo chkconfig --add nagios
error reading information on service nagios: No such file or directory
[ec2-user@ip-172-31-8-148 ~]$ sudo chkconfig nagios on
Note: Forwarding request to 'systemctl enable nagios.service'.
[ec2-user@ip-172-31-8-148 ~]$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
ec2-user@ip-172-31-8-148:~
Website: https://www.nagios.org
Reading configuration data...
   Read main config file okay...
   Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
        Checked 8 services.
        Checked 1 hosts.
        Checked 1 host groups.
        Checked 0 service groups.
        Checked 1 contacts.
        Checked 1 contact groups.
        Checked 24 commands.
        Checked 5 time periods.
        Checked 0 host escalations.
        Checked 0 service escalations.
Checking for circular paths...
        Checked 1 hosts
        Checked 0 service dependencies
        Checked 0 host dependencies
        Checked 5 timeperiods
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
[ec2-user@ip-172-31-8-148 ~]$
```

```
ec2-user@ip-172-31-8-148:~
[ec2-user@ip-172-31-8-148 ~]$ sudo service nagios start
Redirecting to /bin/systemctl start nagios.service
[ec2-user@ip-172-31-8-148 ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.9
     Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
     Active: active (running) since Thu 2024-10-17 09:27:23 UTC; 12min ago
       Docs: https://www.nagios.org/documentation
   Main PID: 70109 (nagios)
      Tasks: 6 (limit: 1112)
     Memory: 6.8M
        CPU: 438ms
     CGroup: /system.slice/nagios.service
             ├─70109 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             ├─70110 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─70111 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─70112 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             ├─70113 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
             └─70114 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Oct 17 09:32:15 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: SERVICE ALERT: localhost;HTTP;WARNING;HARD;4>
Oct 17 09:32:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: SERVICE ALERT: localhost;Swap Usage;CRITICAL>
Oct 17 09:33:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: SERVICE ALERT: localhost;Swap Usage;CRITICAL>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: SERVICE NOTIFICATION: nagiosadmin;localhost;>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: SERVICE ALERT: localhost;Swap Usage;CRITICAL>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: wproc: NOTIFY job 4 from worker Core Worker >
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: wproc:   host=localhost; service=Swap Usage;>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: wproc:   early_timeout=0; exited_ok=1; wait_>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: wproc:   stderr line 01: /bin/sh: line 1: /b>
Oct 17 09:34:45 ip-172-31-8-148.ap-south-1.compute.internal nagios[70109]: wproc:   stderr line 02: /usr/bin/printf: wr>
lines 1-26/26 (END)
[ec2-user@ip-172-31-8-148 ~]$
```



Conclusion:

Thus, we learned about Nagios and successfully set it up on our Linux machine. Nagios proves to be an effective tool for continuous monitoring, helping to detect and resolve issues quickly, ensuring system reliability and smooth operations.