

EXPERIMENT NO. 10

DATE –

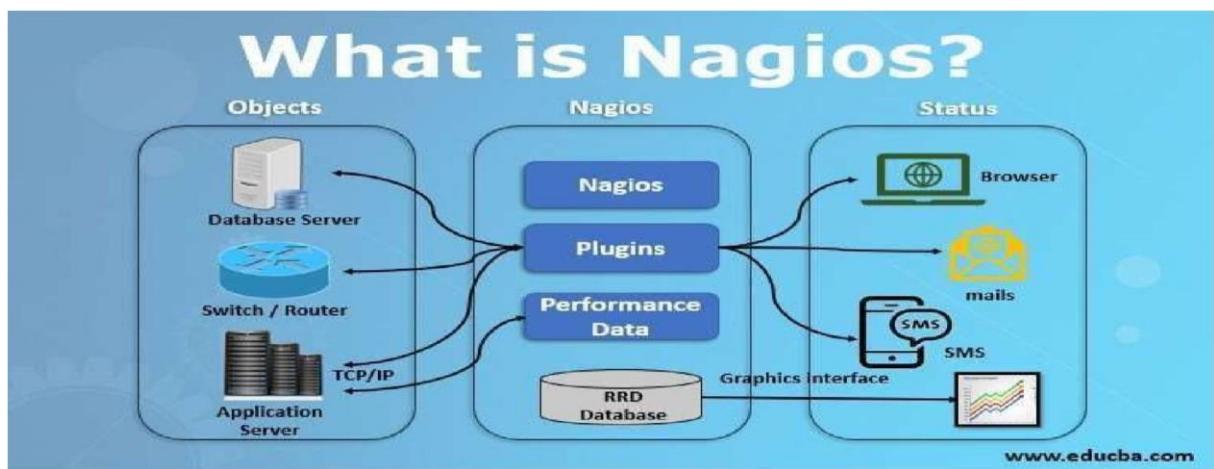
Aim -

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

Theory -

Nagios XI provides a monitoring, alerting, graphing, and reporting platform for your entire infrastructure, including servers, operating systems, applications, network devices, websites, hypervisors, cloud servers, and much more. Thousands of community plugins and the ability to write your own custom plugins enable you to monitor virtually anything.

- The Nagios Core 4 engine provides the powerful, flexible foundation, which Nagios XI makes faster and easier to leverage than ever before.
- Dashboards empower you to create custom visualizations for an at-a-glance view of the data most important to you.
- Alerts can be sent via email or text, in addition to the ability to send an SNMP trap or execute a custom script when problems are found.
- Reports and Graphs enable you to leverage the performance and state history data Nagios XI collects, and can be exported and shared. Reports such as Capacity Planning empower you to project future usage based on historical performance.



Advantages

Implementing effective server monitoring with Nagios offers the following benefits:

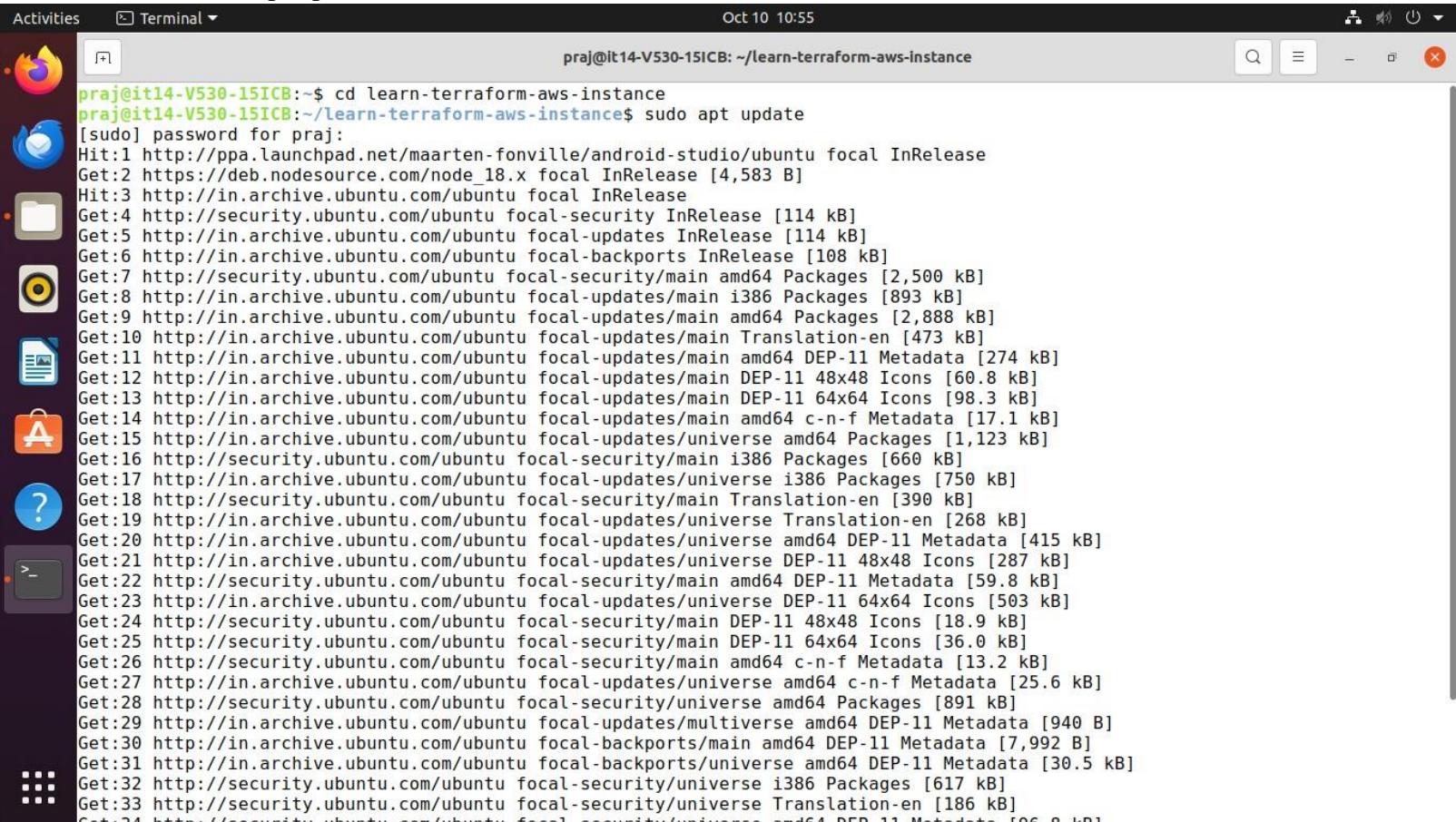
- Increased server, services, process, and application availability
- Fast detection of network and server outages and protocol failures
- Fast detection of failed servers, services, processes and batch jobs

Steps:

Step 1 - Install Packages Dependencies

First, we will update the Ubuntu repository and install some packages dependencies for the Nagios installation. Update the Ubuntu repository using the apt command below.

```
$ sudo apt update
```



The screenshot shows a standard Ubuntu desktop interface with a docked dock on the left containing icons for Dash, Home, Activities, Terminal, and others. A terminal window is open in the center, titled 'Terminal'. The command \$ sudo apt update is run, and the output shows the progress of updating the repository. The terminal window has a dark theme with light-colored text. The status bar at the bottom indicates the date and time as Oct 10 10:55.

```
pra@it14-V530-15ICB:~$ cd learn-terraform-aws-instance
pra@it14-V530-15ICB:~/Learn-terraform-aws-instance$ sudo apt update
[sudo] password for praj:
Hit:1 http://ppa.launchpad.net/maarten-fonville/android-studio/ubuntu focal InRelease
Get:2 https://deb.nodesource.com/node_18.x focal InRelease [4,583 B]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2,500 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [893 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [2,888 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [473 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [274 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-updates/main DEP-11 48x48 Icons [60.8 kB]
Get:13 http://in.archive.ubuntu.com/ubuntu focal-updates/main DEP-11 64x64 Icons [98.3 kB]
Get:14 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [17.1 kB]
Get:15 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [1,123 kB]
Get:16 http://security.ubuntu.com/ubuntu focal-security/main i386 Packages [660 kB]
Get:17 http://in.archive.ubuntu.com/ubuntu focal-updates/universe i386 Packages [750 kB]
Get:18 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [390 kB]
Get:19 http://in.archive.ubuntu.com/ubuntu focal-updates/universe Translation-en [268 kB]
Get:20 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [415 kB]
Get:21 http://in.archive.ubuntu.com/ubuntu focal-updates/universe DEP-11 48x48 Icons [287 kB]
Get:22 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [59.8 kB]
Get:23 http://in.archive.ubuntu.com/ubuntu focal-updates/universe DEP-11 64x64 Icons [503 kB]
Get:24 http://security.ubuntu.com/ubuntu focal-security/main DEP-11 48x48 Icons [18.9 kB]
Get:25 http://security.ubuntu.com/ubuntu focal-security/main DEP-11 64x64 Icons [36.0 kB]
Get:26 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [13.2 kB]
Get:27 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 c-n-f Metadata [25.6 kB]
Get:28 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [891 kB]
Get:29 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:30 http://in.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [7,992 B]
Get:31 http://in.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30.5 kB]
Get:32 http://security.ubuntu.com/ubuntu focal-security/universe i386 Packages [617 kB]
Get:33 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [186 kB]
Get:34 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [0 kB]
```

After this, install packages dependencies for Nagios installation.

```
$ sudo apt install -y autoconf bc gawk dc build-essential gcc libc6 make wget unzip apache2 php libapache2mod-php libgd-dev libmcrypt-dev make libssl-dev snmp libnet-snmp-perl gettext
```

```
praj@it14-V530-15ICB:~/learn-terraform-aws-instance$ sudo apt install -y autoconf bc gawk dc build-essential gcc libc6 make wget unzip apache2 php libapache2- mod-php libgd-dev libmcrypt-dev make libssl-dev snmp libnet-snmp-perl gettext  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
E: Unable to locate package libapache2  
E: Unable to locate package mod-php
```

Step 2 - Install Nagios Core 4.4.6

In this step, we will install the latest stable version Nagios Core 4.4.6. And we will install it manually from the source.

Go to your home directory and download the Nagios Core source code.

```
$ cd ~
```

```
$ wget https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
```

```
praj@it14-V530-15ICB:~/Learn-terraform-aws-instance$ cd ~/
```

```
praj@it14-V530-15ICB:~$ wget https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz  
--2023-10-10 11:35:17-- https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz  
Resolving github.com (github.com)... 20.207.73.82  
Connecting to github.com (github.com)|20.207.73.82|:443... connected.  
HTTP request sent, awaiting response... 302 Found  
Location: https://codeload.github.com/NagiosEnterprises/nagioscore/tar.gz/refs/tags/nagios-4.4.6 [following]  
--2023-10-10 11:35:17-- https://codeload.github.com/NagiosEnterprises/nagioscore/tar.gz/refs/tags/nagios-4.4.6  
Resolving codeload.github.com (codeload.github.com)... 20.207.73.88  
Connecting to codeload.github.com (codeload.github.com)|20.207.73.88|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: unspecified [application/x-gzip]  
Saving to: 'nagios-4.4.6.tar.gz'  
  
nagios-4.4.6.tar.gz.1 [ => ] 10.81M 7.26MB/s in 1.5s  
  
2023-10-10 11:35:19 (7.26 MB/s) - 'nagios-4.4.6.tar.gz.1' saved [11333431]
```

```
praj@it14-V530-15ICB:~$ 
```

Extract the Nagios package and go to the extracted Nagios directory.

```
$ tar -xf nagios-4.4.6.tar.gz
```

```
$ cd nagioscore-*/
```

```
aachal@AACHAL:~$ tar -xf nagios-4.4.6.tar.gz  
aachal@AACHAL:~$ cd nagioscore-*/  
aachal@AACHAL:~/nagioscore-nagios-4.4.6$ |
```

- Compile and Install Nagios

First, compile Nagios source code and define the Apache virtual host configuration for Nagios.

```
$ sudo ./configure --with-httpd-conf=/etc/apache2/sitesenabled $ sudo make all
```

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo ./configure --with-httpd-conf=/etc/apache2/sitesenabled $ sudo
configure: WARNING: you should use --build, --host, --target
configure: WARNING: invalid host type: $
configure: WARNING: you should use --build, --host, --target
configure: checking for a BSD-compatible install... /usr/bin/install -c
configure: checking build system type... Invalid configuration '$': machine '$' not recognized
configure: error: /bin/bash ./config.sub $ failed
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ 

cd ./base && make
make[1]: Entering directory '/home/ubuntu/nagioscore-nagios-4.4.6/base'
gcc -Wall -I... -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
nagios.c: In function 'main':
nagios.c:611:4: warning: ignoring return value of 'asprintf', declared with attribute warn_unused_result [-Wunused-result]
  611 |     asprintf(&mac->x[MACRO_PROCESSSTARTTIME], "%llu", (unsigned long long )program_start);
      |
nagios.c:841:4: warning: ignoring return value of 'asprintf', declared with attribute warn_unused_result [-Wunused-result]
  841 |     asprintf(&mac->x[MACRO_EVENTSTARTTIME], "%llu", (unsigned long long)event_start);
      |
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$
```

Create the Nagios user and group, and add the 'www-data' Apache user to the 'nagios' group.

```
$ sudo make install-groups-users
```

```
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... Invalid configuration '$': machine '$' not recognized
configure: error: /bin/bash ./config.sub $ failed
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-groups-users
```

```
Group nagios already exists
User nagios already exists
```

```
$ sudo usermod -a -G nagios www-data
```

```
sudo apt install udo
```

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo usermod -a -G nagios www-data
usermod: group 'nagios' does not exist
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$
```

Install Nagios binaries, service daemon script, and the command mode \$ sudo make
install

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo usermod -a -G nagios www-data
usermod: group 'nagios' does not exist
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install
sudo: make: command not found
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ cd ./base && make install
make[1]: Entering directory '/home/ubuntu/nagioscore-nagios-4.4.6/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/ubuntu/nagioscore-nagios-4.4.6/base'
cd ./cgi && make install
make[1]: Entering directory '/home/ubuntu/nagioscore-nagios-4.4.6/cgi'
make install-basic
make[2]: Entering directory '/home/ubuntu/nagioscore-nagios-4.4.6/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; \

```

```
$ sudo make install-daemoninit
```

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install
sudo: make: command not found
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-daemoninit
```

```
/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
*** Init script installed ***
$ sudo make install-commandmode
```

```
sudo: make: command not found
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-commandmode
sudo: make: command not found
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-config
```

```
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw  
chmod g+s /usr/local/nagios/var/rw  
  
*** External command directory configured ***
```

After that, install the sample script configuration.

```
$ sudo make install-config
```

```
sudo: make: command not found  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-config  
sudo: make: command not found  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ □
```

```
/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/loc  
al/nagios/etc/cgi.cfg  
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /us  
r/local/nagios/etc/resource.cfg  
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/  
templates.cfg /usr/local/nagios/etc/objects/templates.cfg  
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/  
commands.cfg /usr/local/nagios/etc/objects/commands.cfg
```

Then install the Apache configuration for Nagios and activate the mod_rewrite and mode_cgi modules.

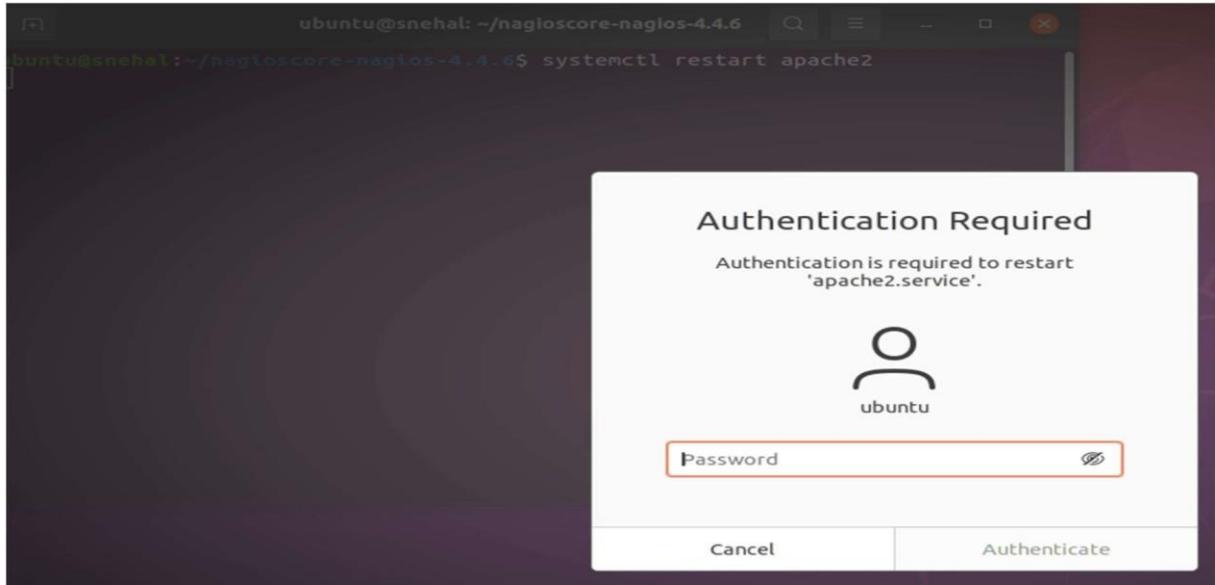
```
sudo: make: command not found  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo make install-webconf  
sudo: make: command not found  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ □  
$ sudo make install-webconf
```

```
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-enabled/n  
agios.conf  
if [ 0 -eq 1 ]; then \  
    ln -s /etc/apache2/sites-enabled/nagios.conf /etc/apache2/sites-enabled/  
nagios.conf; \  
fi  
  
*** Nagios/Apache conf file installed ***
```

```
$ sudo a2enmod rewrite cgi
```

Now restart the Apache service.

```
$ systemctl restart apache2
```



And we have installed the Nagios Core 4.4.6.

- Create nagiosadmin user

After installing the Nagios Core, we will add the basic authentication for accessing the Nagios dashboard. And we will be using the basic Apache authentication.

Create a new apache basic authentication for the user the "nagiosadmin".
\$ sudo
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin Type your strong
password.



And we have created a new user 'nagiosadmin' for the Nagios dashboard authentication.

- Setup UFW Firewall

For the firewall configuration, you will need to add the Apache service and the Nagios server port to the UFW firewall.

Add the SSH and Apache HTTP port using the ufw command below.

```
$ for svc in Apache ssh do  
sudo ufw allow $svc done
```

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ for svc in Apache ssh do  
> do  
> sudo ufw allow $svc  
> done  
ERROR: Could not find a profile matching 'Apache'  
Skipping adding existing rule  
Skipping adding existing rule (v6)  
ERROR: Could not find a profile matching 'do'  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$
```

Next, start the UFW firewall service and add it to the system boot.

```
$ sudo ufw enable
```

```
ERROR: Could not find a profile matching 'do'  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo ufw enable  
Firewall is active and enabled on system startup  
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$
```

Type 'y' and the UFW firewall service will be activate.

Now check all available rules using the command below.

```
$ sudo ufw status numbered
```

Now we will get both the SSH and Apache services added to the UFW firewall.

Status: active			
To	Action	From	
--	-----	-----	
[1] Apache	ALLOW IN	Anywhere	
[2] 22/tcp	ALLOW IN	Anywhere	
[3] Apache (v6)	ALLOW IN	Anywhere (v6)	
[4] 22/tcp (v6)	ALLOW IN	Anywhere (v6)	

And finally, you've completed the Nagios Core installation on the Ubuntu 20.04 server.

Step 3 - Install Nagios Plugins and NRPE Plugin

After installing the Nagios Core, we will install the Nagios Plugins and NRPE Plugins.

Both Nagios and NRPE plugins are available by default on the Ubuntu repository. You can install those packages using the apt command below.

```
$ sudo apt install monitoring-plugins nagios-nrpe-plugin
```

```
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo apt install monitoring-plugins nagios-nrpe-plugin
Reading package lists... Done
Building dependency tree
Reading state information... Done
nagios-nrpe-plugin is already the newest version (4.0.0-2ubuntu1).
monitoring-plugins is already the newest version (2.2-6ubuntu1.2).
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 libfwupdplugin1 libxmlb1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 13 not upgraded.
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$
```

Once the installation is complete, go to the nagios installation directory "/usr/local/nagios/etc" and create a new directory for storing all server hosts configuration. \$ cd /usr/local/nagios/etc

```
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 13 not upgraded.
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ cd /usr/local/nagios/etc
bash: cd: /usr/local/nagios/etc: No such file or directory
```

```
$ mkdir -p /usr/local/nagios/etc/servers
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 13 not upgraded.
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ cd /usr/local/nagios/etc
bash: cd: /usr/local/nagios/etc: No such file or directory
```

Next, edit the Nagios configuration 'nagios.cfg' using vim editor. \$

```
sudo nano nagios.cfg
```

```
bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/servers
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo nano nagios.cfg
```

Uncomment the 'cfg_dir' option that will be used for storing all server hosts configurations.

```
cfg_dir=/usr/local/nagios/etc/servers \
```

```
# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

# Definitions for monitoring a network printer
#cfg_file=/usr/local/nagios/etc/objects/printer.cfg

# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:
```

```
cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
```

**^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^L Go To Line**

Save and close.

Next edit the configuration file "resource.cfg" and define the path binary files of Nagios Monitoring Plugins.

```
$ sudo nano resource.cfg
```

```
bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/servers
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo nano resource.cfg
```

Define the Nagios Monitoring Plugins path by changing the default configuration as below.

```
$USER1$=/usr/lib/nagios/plugins
```

```
GNU nano 4.8          resource.cfg
# event handlers - if you decide to move the plugins or event handlers to
# a different directory in the future, you can just update one or two
# $USERx$ macros, instead of modifying a lot of command definitions.
#
# The CGIs will not attempt to read the contents of resource files, so
# you can set restrictive permissions (600 or 660) on them.
#
# Nagios supports up to 256 $USERx$ macros ($USER1$ through $USER256$)
#
# Resource files may also be used to store configuration directives for
# external data sources like MySQL...
#
#####
#
# Sets $USER1$ to be the path to the plugins
$USER1$=/usr/lib/nagios/plugins

# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/local/nagios/libexec/eventhandlers
#
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell ^L Go To Line
```

Save and close.

After that, add the nagios admin email contacts by editing the configuration file "objects/contacts.cfg".
\$ sudo nano objects/contacts.cfg

```
bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/servers
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo nano objects/contacts.cfg
```

Change the email address with your own.

```

GNU nano 4.8          objects/contacts.cfg          Modified
# #####
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.

define contact {

    contact_name      nagiosadmin           ; Short name of user
    use               generic-contact        ; Inherit default values fr>
    alias             Nagios Admin         ; Full name of user
    email             raisnehalonline@gmail.com ; <<***** CHANGE THIS TO >>
}

#####
# CONTACT GROUPS

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File   ^\ Replace   ^U Paste Text^T To Spell  ^L Go To Line

```

Save and close.

Now define the nrpe check command by editing the configuration file

"objects/commands.cfg". \$ sudo nano objects/commands.cfg

```

bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo nano objects/commands.cfg

```

Add the following configuration to the end of the line.

```

#####
define command {
    command_name      process-host-perfdata
    command_line      /usr/bin/printf "%b" "$LASTHOSTCHECKS\$HOSTNAME\$HOSTSTATUS"
}

define command {
    command_name      process-service-perfdata
    command_line      /usr/bin/printf "%b" "$LASTSERVICECHECKS\$HOSTNAME\$SERVICESTATU"
}
define command{
    command_name check_nrpe
    command_line $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$
}

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File   ^\ Replace   ^U Paste Text^T To Spell  ^L Go To Line

```

Save and close, and the Nagioscore configuration has been completed.

Next, start the Nagios service and add it to the system boot. \$ sudo
systemctl start nagios

```
bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/servers
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo systemctl start nagios
```

\$ sudo systemctl enable nagios

\$ sudo systemctl status nagios

The Nagios service is up and running, check using the following command.

```
aachal@aachal:~/nagioscore-nagios-4.4.6$ sudo systemctl status nagios
```

```
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2021-10-02 18:03:41 IST; 38min ago
     Docs: https://www.nagios.org/documentation
 Main PID: 971 (nagios)
    Tasks: 8 (limit: 4505)
   Memory: 4.9M
      CPU: 0.000 CPU(s) (idle)
     CGroup: /system.slice/nagios.service
             └─971 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
                 ├─972 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 ├─973 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 ├─974 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 ├─975 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 ├─976 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 ├─977 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/...
                 └─983 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
```

As a result, the Nagios service is up and running. Now we need to restart the Apache service to apply a new Nagios configuration.

\$ sudo systemctl restart apache2

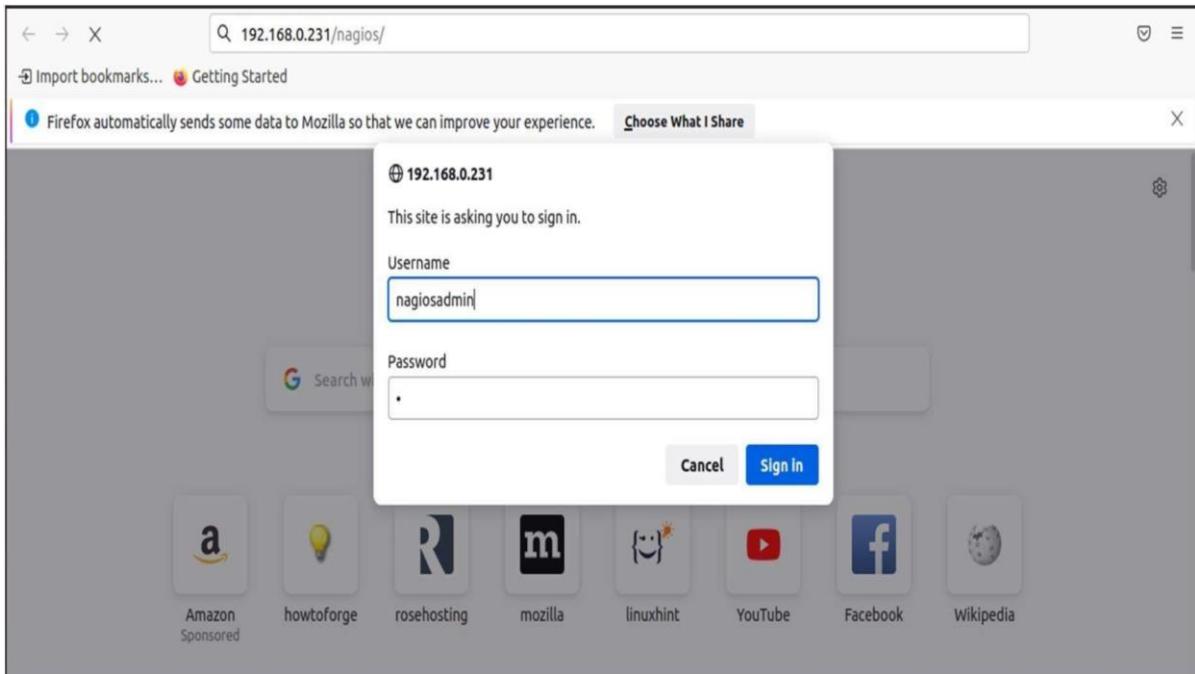
```
bash: cd: /usr/local/nagios/etc: No such file or directory
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ mkdir -p /usr/local/nagios/etc/servers
mkdir: cannot create directory '/usr/local/nagios': Permission denied
praj@it14-V530-15ICB:~/nagioscore-nagios-4.4.6$ sudo systemctl restart apache2
```

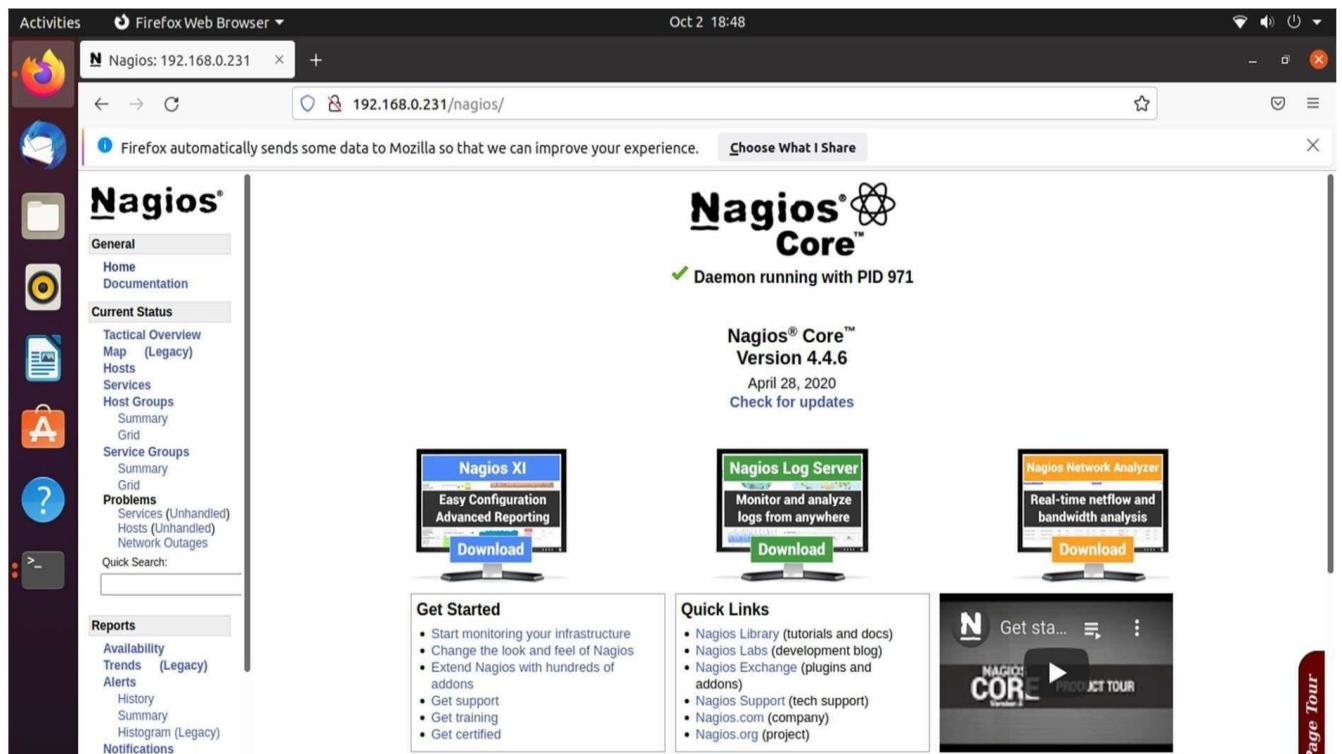
And the Nagios configuration has been completed.

Open your web browser and type the server IP address following the "nagios" URL path.

<http://your ip add/nagios/>

Log in with the user "nagiosadmin" and type your password





Step 5 - Add Linux Host to Monitor

In this step, we will add the Ubuntu server with hostname "client01" and the IP address "192.27.0.12" to the Nagios server.

- Install NRPE Server on the Client01 Server

Log in to the "client01" server using your ssh.

```
$ sudo ssh root@192.27.0.12
```

Once we have logged in, update the Ubuntu repository and install Nagios Plugins and NRPE Server.

```
$ sudo apt update
```

```
$ sudo apt install nagios-nrpe-server monitoring-plugins
```

Activities Terminal Oct 10 11:50

```
praj@it14-V530-15ICB:~$ sudo apt install nagios-nrpe-server monitoring-plugins
[sudo] password for praj:
Reading package lists... Done
Building dependency tree
Reading state information... Done
monitoring-plugins is already the newest version (2.2-6ubuntu1.2).
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 libfwupdplugin1 libxml2
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  xinetd | inetd
The following NEW packages will be installed:
  nagios-nrpe-server
0 upgraded, 1 newly installed, 0 to remove and 13 not upgraded.
Need to get 359 kB of archives.
After this operation, 476 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 nagios-nrpe-server amd64 4.0.0-2ubuntu1 [359 kB]
Fetched 359 kB in 1s (295 kB/s)
Selecting previously unselected package nagios-nrpe-server.
(Reading database ... 188607 files and directories currently installed.)
Preparing to unpack .../nagios-nrpe-server_4.0.0-2ubuntu1_amd64.deb ...
Unpacking nagios-nrpe-server (4.0.0-2ubuntu1) ...
Setting up nagios-nrpe-server (4.0.0-2ubuntu1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nagios-nrpe-server.service → /lib/systemd/system/nagios-nrpe-server.service.
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for systemd (245.4-4ubuntu3.22) ...
praj@it14-V530-15ICB:~$ 
```

Next, go to the NRPE installation directory "/etc/nagios" and edit the configuration file "nrpe.cfg". \$ cd /etc/nagios/

```
Processing triggers for systemd (245.4-4ubuntu3.22) ...
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ cd /etc/nagios/
```

```
$ Sudo nano nrpe.cfg
PROCESSING TRIGGERS FOR SYSTEMD (245.4-4ubuntu3.22) ...
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ sudo nano nrpe.cfg
```

Uncomment the "server_address" line and change the value with the "client01" IP address.
server_address=192.27.0.12

On the "allowed_hosts" line, add the Nagios Server IP address "172.16.0.5".

allowed_hosts=127.0.0.1,::1,172.16.0.5

The image shows two terminal windows side-by-side, both displaying the nano text editor interface. The top window shows the configuration for the NRPE daemon, specifically the 'nrpe.cfg' file. It contains settings for the port number (server_port=5666) and the server address (server_address=127.0.0.1). The bottom window shows the configuration for the Nagios daemon, specifically the 'nrpe_local.cfg' file. It includes a section for allowed host addresses, which is currently empty. Both windows have a menu bar at the top with options like Get Help, Write Out, Where Is, Cut Text, Justify, Cur Pos, Exit, Read File, Replace, Paste Text, To Spell, and Go To Line.

```
# PORT NUMBER
# Port number we should wait for connections on.
# NOTE: This must be a non-privileged port (i.e. > 1024).
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

server_port=5666

# SERVER ADDRESS
# Address that nrpe should bind to in case there are more than one interface
# and you do not want nrpe to bind on all interfaces.
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

server_address=127.0.0.1

nrpe_group=nagios

# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon. Network addresses with a bit mask
# (i.e. 192.168.1.0/24) are also supported. Hostname wildcards are not currently
# supported.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address. I would highly recommend adding entries in your /etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
```

Save and close.

Next, edit the "nrpe_local.cfg" configuration.

Change the IP address with the "client01" IP address, and paste the configuration into it.

```
GNU nano 4.8                               nrpe_local.cfg                         Modified
#####
# Do any local nrpe configuration here
#####
command[check_root]=/usr/lib/nagios/plugins/check_disk -w 20% -c 10% -p /
command[check_ping]=/usr/lib/nagios/plugins/check_ping -H 192.168.0.231 -w 100.>
command[check_ssh]=/usr/lib/nagios/plugins/check_ssh -4 192.168.0.231
command[check_http]=/usr/lib/nagios/plugins/check_http -I 192.168.0.231
command[check_apt]=/usr/lib/nagios/plugins/check_apt
```

Save and close.

Now restart the NRPE service and add it to the system boot.

```
$ sudo systemctl restart nagios-nrpe-server
```

```
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ sudo systemctl restart nagios-nrpe-server
```

```
$ sudo systemctl enable nagios-nrpe-server
```

And the Nagios NRPE server is up and running.

Check the NRPE service using the following command.

```
$ sudo systemctl status nagios-nrpe-server
```

```
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ sudo systemctl restart nagios-nrpe-server
```

```
● nagios-nrpe-server.service - Nagios Remote Plugin Executor
   Loaded: loaded (/lib/systemd/system/nagios-nrpe-server.service; enabled; v>
   Active: active (running) since Sat 2021-10-02 19:11:14 IST; 1min 18s ago
     Docs: http://www.nagios.org/documentation
 Main PID: 10126 (nrpe)
    Tasks: 1 (limit: 4505)
   Memory: 672.0K
      CGrou: /system.slice/nagios-nrpe-server.service
              └─10126 /usr/sbin/nrpe -c /etc/nagios/nrpe.cfg -f

Oct 02 19:11:14 snehal nrpe[9244]: Daemon shutdown
Oct 02 19:11:14 snehal systemd[1]: Stopping Nagios Remote Plugin Executor...
Oct 02 19:11:14 snehal systemd[1]: nagios-nrpe-server.service: Succeeded.
Oct 02 19:11:14 snehal nrpe[10126]: Starting up daemon
Oct 02 19:11:14 snehal systemd[1]: Started Nagios Remote Plugin Executor.
Oct 02 19:11:14 snehal nrpe[10126]: Server listening on 127.0.0.1 port 5666.
Oct 02 19:11:14 snehal systemd[1]: Started Nagios Remote Plugin Executor.
Oct 02 19:11:14 snehal nrpe[10126]: Listening for connections on port 5666
Oct 02 19:11:14 snehal nrpe[10126]: Allowing connections from: 127.0.0.1,::192.168.1.10
lines 1-19/19 (END)
```

The NRPE service is up and running.

Next, back to the Nagios Server and check the "client01" NRPE server.

```
/usr/lib/nagios/plugins/check_nrpe -H 192.27.0.12
```

```
/usr/lib/nagios/plugins/check_nrpe -H 192.27.0.12 -c check_ping
```

```
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ /usr/lib/nagios/plugins/check_nrpe -H 192.27.0.12 -c check_ping
```

And we have installed the Nagios NRPE Server and Nagios Plugins on the "client01" host.

- Add Hosts Configuration to the Nagios Server

```
$ cd /usr/local/nagios/etc
```

```
$ sudo nano servers/client01.cfg
```

Change the IP address and the hostname with your own and paste the configuration into it.
Back to the Nagios server terminal, go to the "/usr/local/nagios/etc" directory and create a new configuration "server/client01.cfg".

```
# Ubuntu Host configuration file1
```

```
define host {
    use          linux-server
    host_name    client01    alias      Ubuntu
    Host         address     192.27.0.12   register
    1 }
```

Save and close.

Now restart the Nagios Serve

```
$ sudo systemctl restart nagios
[Processing triggers for system (245.4-4ubuntu2) ...]
praj@it14-V530-15ICB:~$ cd
praj@it14-V530-15ICB:~$ sudo systemctl restart nagios
```

Step 5 - Testing

Back to your browser and wait for some minutes.

Click on the "Hosts" menu and you will get the "client01" has been added.

Firefox Web Browser ▾

N Nagios: 192.168.0.231 X How to Install Nagios Mo X Oct 2 19:22

192.168.0.231/nagios/

Nagios®

General

- Home
- Documentation

Current Status

- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
 - Summary
 - Grid
- Service Groups
 - Summary
 - Grid
- Problems
 - Services (Unhandled)
 - Hosts (Unhandled)
 - Network Outages

Quick Search:

Current Network Status

Last Updated: Sat Oct 2 19:22:06 IST 2021
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

Host Status Totals

Up	Down	Unreachable	Pending
1	0	0	0

All Problems All Types

0	1
---	---

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
7	0	0	1	0

All Problems All Types

1	8
---	---

Host Status Details For All Host Groups

Limit Results: 100 ▾

Host	Status	Last Check	Duration	Status Information
localhost	UP	10-02-2021 19:18:42	0d 18h 55m 44s	PING OK - Packet loss = 0%, RTA = 0.07 ms

Results 1 - 1 of 1 Matching Hosts

Activities Firefox Web Browser ▾

N Nagios: 192.168.0.231 X How to Install Nagios Mo X Oct 2 19:23

192.168.0.231/nagios/

Nagios®

General

- Home
- Documentation

Current Status

- Tactical Overview
- Map (Legacy)
- Hosts
- Services
- Host Groups
 - Summary
 - Grid
- Service Groups
 - Summary
 - Grid
- Problems
 - Services (Unhandled)
 - Hosts (Unhandled)
 - Network Outages

Quick Search:

Host Information

Last Updated: Sat Oct 2 19:23:19 IST 2021
Updated every 90 seconds
Nagios® Core™ 4.4.6 - www.nagios.org
Logged in as nagiosadmin

Host

localhost
(localhost)

Member of

linux-servers

127.0.0.1

Host State Information

Host Status: UP (for 0d 18h 56m 57s)

Status Information: PING OK - Packet loss = 0%, RTA = 0.07 ms

Performance Data: rta=0.066000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0

Current Attempt: 1/10 (HARD state)

Last Check Time: 10-02-2021 19:18:42

Check Type: ACTIVE

Check Latency / Duration: 0.000 / 4.128 seconds

Next Scheduled Active Check: 10-02-2021 19:23:42

Last State Change: 10-02-2021 00:26:22

Last Notification: N/A (notification 0)

Is This Host Flapping? NO (0.00% state change)

In Scheduled Downtime? NO

Last Update: 10-02-2021 19:23:14 (0d 0h 0m 5s ago)

Active Checks: ENABLED

Passive Checks: ENABLED

Obsessing: ENABLED

Notifications: ENABLED

Event Handler: ENABLED

Flap Detection: ENABLED

Host Commands

- Locate host on map
- Disable active checks of this host
- Re-schedule the next check of this host
- Submit passive check result for this host
- Stop accepting passive checks for this host
- Stop obsessing over this host
- Disable notifications for this host
- Send custom host notification
- Schedule downtime for this host
- Schedule downtime for all services on this host
- Disable notifications for all services on this host
- Enable notifications for all services on this host
- Schedule a check of all services on this host
- Disable checks of all services on this host
- Enable checks of all services on this host
- Disable event handler for this host
- Disable flap detection for this host
- Clear flapping state for this host

Host Comments

Add a new comment Delete all comments

Entry Time Author Comment Comment ID Persistent Type Expires Actions

This host has no comments associated with it

Page Tour

Conclusion -

Nagios Log Server greatly simplifies the process of searching your log data. Set up alerts to notify you when potential threats arise, or simply filter your data to quickly audit your system. With Log Server, you get all of your data in one location, with high availability and fail-over built right in. Thus, with all commands we have installed Nagios on Ubuntu servers.

