

# Nagios Core Installation And Configuration

## Step 1: Update Your System

Before installing any new software, update your system packages.

```
# sudo yum update
```

```
[root@localhost ~]# yum update -y
```

## Step 2: Install Required Packages

Nagios Core requires several packages to function correctly. Install them using the following command.

```
# sudo yum install gcc glibc glibc-common wget unzip httpd php gd gd-devel perl postfix
```

```
[root@localhost ~]# sudo yum install gcc glibc glibc-common wget unzip httpd php gd gd-devel perl postfix -y
```

```
# dnf -y groupinstall "Development Tools"
```

```
[root@localhost ~]# dnf -y groupinstall "Development Tools"
```

## Step 3: Download Nagios Source File

Use the following commands to download the Nagios source file.

```
# mkdir /nagios && cd /nagios
```

```
# wget
```

```
https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.6.tar.gz
```

```
[root@localhost ~]# mkdir /nagios && cd /nagios
[root@localhost nagios]# sudo wget -O nagioscore.tar.gz https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.5.tar.gz
--2024-08-29 08:46:53-- https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.5.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-4.4.5 [following]
--2024-08-29 08:46:59-- https://codeload.github.com/NagiosEnterprises/nagioscore/tar.gz/refs/tags/nagios-4.4.5
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: 'nagioscore.tar.gz'

nagioscore.tar.gz           [          ]  <=>

2024-08-29 08:47:10 (1.12 MB/s) - 'nagioscore.tar.gz' saved [11304640]

[root@localhost nagios]# ls
nagioscore.tar.gz
```

```
# sudo tar xzvf nagioscore.tar.gz
```

```
[root@localhost nagios]# tar xzvf nagioscore.tar.gz
nagioscore-nagios-4.4.5/
```

## Step 4: Compile Nagios

After a successful download of the Nagios source file, you can run the commands below to compile Nagios on Almalinux to make sure all dependencies on your system are available.

```
# cd nagioscore-nagios-4.4.5
```

```
# ./configure
```

```
[root@localhost nagioscore-nagios-4.4.5]# ./configure
```

```
*** Configuration summary for nagios 4.4.5 2019-08-20 ***:
```

### General Options:

```
-----
Nagios executable:  nagios
Nagios user/group:  nagios,nagios
Command user/group: nagios,nagios
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:       /usr/sbin/sendmail
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):
```

```
Review the options above for accuracy.  If they look okay,
type 'make all' to compile the main program and CGIs.
```

```
# sudo make all
```

```
[root@localhost nagioscore-nagios-4.4.5]# make all
```

\*\*\* Support Notes \*\*\*\*\*

If you have questions about configuring or running Nagios, please make sure that you:

- Look at the sample config files
- Read the documentation on the Nagios Library at:  
<https://library.nagios.com>

before you post a question to one of the mailing lists.

### Step 5 :Create Nagios User and Group on Almalinux

# sudo make install-groups-users

```
[root@localhost nagios-4.4.6]# make install-groups-users
groupadd -r nagios
useradd -g nagios nagios
[root@localhost nagios-4.4.6]# usermod -a -G nagios apache
```

### Step 6: Add Apache User

To the user you created in the previous step, you can add an Apache user by running the command below

# sudo usermod -a -G nagios apache

## Step 7: Install Nagios Binaries, HTML, and CGIs Files

# sudo make install

```
*** Main program, CGIs and HTML files installed ***

You can continue with installing Nagios as follows (type 'make'
without any arguments for a list of all possible options):

make install-init
    - This installs the init script in /lib/systemd/system

make install-commandmode
    - This installs and configures permissions on the
      directory for holding the external command file

make install-config
    - This installs sample config files in /usr/local/nagios/etc

make[1]: Leaving directory '/nagios/nagios-4.4.6'
```

## Step 8: Install Daemon and Enable the Service to Start on Boot, To install Command mode, Install Sample Config File, Install Apache Config Files

# sudo make install-daemoninit

```
[root@localhost nagios-4.4.6]# 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
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /usr/lib/systemd/system/nagios.service.

*** Init script installed ***
```

#sudo make install-commandmode

```
[root@localhost nagios-4.4.6]# make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
```

#sudo make install-config

```
[root@localhost nagios-4.4.6]# make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/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 ***
```

```
# sudo make install-webconf
```

```
[root@localhost nagios-4.4.6]# 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 ***
```

## Step 9: Configure Firewall

In this step, you are ready to configure the firewall and open port 80 for inbound traffic.

```
# sudo firewall-cmd --zone=public --add-port=80/tcp
# sudo firewall-cmd --zone=public --add-port=80/tcp --permanent
# sudo firewall-cmd --reload
```

```
[root@localhost nagios-4.4.6]# sudo firewall-cmd --zone=public --add-port=80/tcp
success
[root@localhost nagios-4.4.6]# sudo firewall-cmd --zone=public --add-port=80/tcp --permanent
success
[root@localhost nagios-4.4.6]# sudo firewall-cmd --reload
success
```

## Step 10: Create Nagios Web login user

The command below lets you create a Nagios web login user.

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

```
[root@localhost nagios-4.4.6]# sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
```

## Step 11: Chekck the nagios installation and configuration command

```
# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

```
[root@localhost nagios-4.4.6]# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Reading configuration data...
  Read main config file okay...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking objects...
  Checked 8 services.
  Checked 1 hosts.
  Checked 1 host groups.
  Checked 0 service groups.
  Checked 1 contacts.
  Checked 1 contact groups.
  Checked 24 commands.
  Checked 5 time periods.
  Checked 0 host escalations.
  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
```

## Step 12: Install Nagios Core Plugins

The next step is to install Nagios plugins. These are standalone extensions that process command-line arguments and monitor just about anything in Nagios Core.

```
# wget https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz
# tar zxvf nagios-plugins-2.4.11.tar.gz
# ./configure
# make
# make install
# chown -R nagios:nagios /usr/local/nagios/libexec/
# chmod -R 755 /usr/local/nagios/libexec/
```

## Step 13: Disable selinux

First, you need to disable SELinux On CentOS. To do this temporarily, run. However, you can disable SELinux permanently. Simply run.

```
# setenforce 0
```

```
# sudo sed -i 's/SELINUX=.*/SELINUX=disabled/g' /etc/selinux/config
```

## Step 14: Restart the Apache & nagios services

```
# systemctl restart httpd
```

```
# systemctl restart nagios
```

## Step 15: Nagios configuration file path

```
# /usr/local/nagios/etc → Config File path
```

```
# /usr/local/nagios/var → Log File path
```

The screenshot displays the Nagios web interface. At the top, the browser address bar shows '10.10.11.175/nagios/'. The interface includes a sidebar with navigation links such as 'General', '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', 'Reports', 'Availability', and 'Trends (Legacy)'. The main content area is divided into several sections: 'Current Network Status' (Last Updated: Tue Aug 29 11:55:33 IST 2024), 'Host Status Totals' (Up: 1, Down: 0, Unreachable: 0, Pending: 0), and 'Service Status Totals' (Ok: 7, Warning: 1, Unknown: 0, Critical: 0, Pending: 0). Below these, there is a 'Service Status Details For All Hosts' table. The table has columns for 'Host', 'Service', 'Status', 'Last Check', 'Duration', 'Attempt', and 'Status Information'. The 'localhost' host is listed with services like 'Current Load', 'Current Users', 'HTTP', 'PING', 'Root Partition', 'SSH', 'Swap Usage', and 'Total Processes'. The 'HTTP' service is highlighted in yellow with a 'WARNING' status. The 'Status Information' column provides detailed messages for each service, such as 'OK - load average: 0.01, 0.05, 0.26' for 'Current Load' and 'HTTP WARNING: HTTP/1.1 403 Forbidden - 4956 bytes in 0.001 second response time' for 'HTTP'.

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	08-29-2024 11:52:06	0d 0h 3m 27s	1/4	OK - load average: 0.01, 0.05, 0.26
	Current Users	OK	08-29-2024 11:52:44	0d 0h 3m 49s	1/4	USERS OK - 5 users currently logged in
	HTTP	WARNING	08-29-2024 11:53:21	0d 0h 2m 12s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 4956 bytes in 0.001 second response time
	PING	OK	08-29-2024 11:53:59	0d 0h 1m 34s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
	Root Partition	OK	08-29-2024 11:54:36	0d 0h 5m 57s	1/4	DISK OK - free space: / 64824 MB (90.51% inode=99%):
	SSH	OK	08-29-2024 11:55:14	0d 0h 5m 19s	1/4	SSH OK - OpenSSH_8.7 (protocol 2.0)
	Swap Usage	OK	08-29-2024 11:50:51	0d 0h 4m 42s	1/4	SWAP OK - 100% free (7999 MB out of 7999 MB)
	Total Processes	OK	08-29-2024 11:51:29	0d 0h 4m 4s	1/4	PROCS OK: 52 processes with STATE = RSZDT

## Nagios Email Installation And Configuration

### Step 1: Install Postfix & s-nail

You need an MTA like Postfix or Sendmail to send email notifications. For this example, we'll use Postfix & s-nail.

```
# sudo dnf install s-nail postfix -y
```

```
[root@localhost html]# sudo dnf install s-nail postfix -y
Last metadata expiration check: 17:31:31 ago on Wed 28 Aug 2024 06:52:25 PM IST.
Dependencies resolved.
=====
Package                                Architecture
=====
Installing:
postfix                                x86_64
s-nail                                x86_64
Transaction Summary
=====
Install 2 Packages
```

```
# sudo systemctl start postfix
```

```
# sudo systemctl enable postfix
```



## Step 2: Configure Postfix

If you need to send emails through an external SMTP server, you'll need to configure Postfix accordingly. Edit the Postfix configuration.

```
# sudo nano /etc/postfix/main.cf
```

### 1. Add the following lines.

```
relayhost = [smtp.gmail.com]:587
smtp_sasl_auth_enable = yes
smtp_sasl_security_options = noanonymous
smtp_sasl_password_maps = hash:/etc/postfix/sasl/sasl_passwd
smtp_tls_security_level = verify
smtp_tls_CAfile = /etc/ssl/certs/ca-bundle.crt
```

```
#
#smtp_tls_CAfile = /etc/pki/tls/certs/ca-bundle.crt

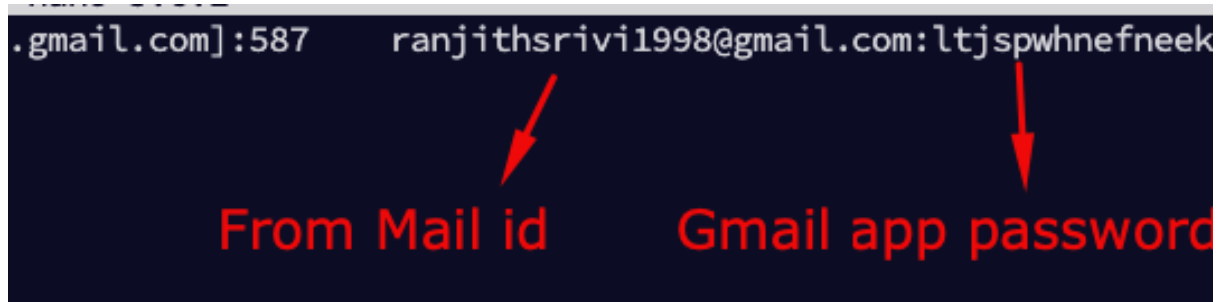
# Use TLS if this is supported by the remote SMTP server, otherwise use
# plaintext (opportunistic TLS outbound).
#
relayhost = [smtp.gmail.com]:587
smtp_sasl_auth_enable = yes
smtp_sasl_security_options = noanonymous
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_tls_security_level = verify
smtp_tls_CAfile = /etc/ssl/certs/ca-bundle.crt
```

### 2. Create the sasl\_passwd file with your SMTP credentials

```
# sudo nano /etc/postfix/sasl_passwd
```

```
[smtp.gmail.com]:587 ranjithsrivi1998@gmail.com:ltjspwhnefneekfa
```

```
.gmail.com]:587    ranjithsrivi1998@gmail.com:ltjspwhnefneekfa
```



From Mail id      Gmail app password

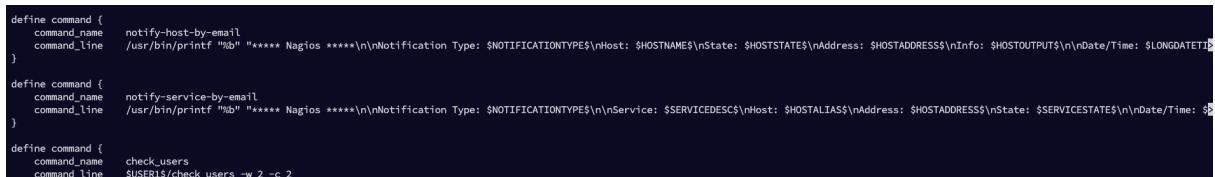
### Step 3: Configure Nagios for Email Notifications

Open And add the file where Nagios commands.cfg

```
# nano /usr/local/nagios/etc/objects/commands.cfg
```

```
define command {
    command_name    notify-host-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState:
$HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo:
$HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME>
}
```

```
define command {
    command_name    notify-service-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost:
$HOSTALIAS$\nAddress: $HOSTADDRESS$\nState:
$SERVICESTATE$\n\nDate/Time: $>
}
```



```
define command {
    command_name    notify-host-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME>
}

define command {
    command_name    notify-service-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type: $NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\nState: $SERVICESTATE$\n\nDate/Time: $
}

define command {
    command_name    check_users
    command_line    $USER1$/check_users -w 2 -c 2
}
```

### Step 4: Configure Contacts

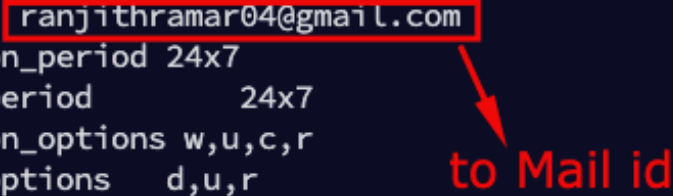
Edit the contacts configuration file

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

```
define contact {
    contact_name    nagiosadmin
    alias           Nagios Admin
    email           ranjithramar04@gmail.com
    service_notification_period 24x7
    host_notification_period    24x7
    service_notification_options w,u,c,r
    host_notification_options  d,u,r
}
```

```
service_notification_commands notify-service-by-email
host_notification_commands notify-host-by-email
}
```

```
t {
  ct_name      nagiosadmin
               Nagios Admin
               ranjithramar04@gmail.com
  ce_notification_period 24x7
  notification_period    24x7
  ce_notification_options w,u,c,r
  notification_options    d,u,r
  ce_notification_commands notify-service-by-email
  notification_commands  notify-host-by-email
}
```



to Mail id

#### Step 5: Nagios configuration file check and restart the service

Restart the Nagios service to apply the changes

```
# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
# sudo systemctl restart nagios
```

#### Step 6: Manually mail check and conformation

```
# echo -e "To: sysad@trioangle.com\nSubject: Test Email\n\nThis is the loose  
of the email" | /usr/sbin/sendmail -t
```