NAGIOS INSTALLATION ON UBUNTU

Nagios installation has been divided into three parts.

1) Installing Nagios on Master
2) Installing NRPE on slave
3) Installing Check_NRPE Plugin on Master

Commands with green font represents master, white font terminal represents slave.

Installing Nagios on Master

Step 1: Update package index.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo apt-get update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [8
8.7 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease
[74.6 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/restricted Sources [
5324 B]
Get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/universe Sources [90
51 kB]
Get:6 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic/multiverse Sources [
181 kB]
```

Step 2: Run the following two commands after that.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo apt-get install wget build-essential unzip openssl libssl-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
wget is already the newest version (1.19.4-1ubuntu2.1).
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-7 dpkg-dev fakeroot g++ g++-7 gcc
  gcc-8-base libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan4
  libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libdpkg-perl libfakeroot libfile-fcntllock-perl
  libgomp1 libisl19 libitm1 liblsan0 libmpc3 libmpx2 libquadmath0 libssl-doc libssl1.1 libstdc
  libtsan0 libubsan0 linux-libc-dev make manpages-dev
```

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo apt-get install apache2 php libapache2-mod-php php-gd libgd-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils fontconfig-config fonts-dejavu-core libapache2-mod-php7
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libexpat1-dev libfontconfig1 libfontconfig1-dev libfr
  libice-dev libice6 libjbig-dev libjbig0 libjpeg-dev libjpeg-turbo8 libjpeg-turbo8-dev libjpeg8
  liblua5.2-0 liblzma-dev libpng-dev libpng-tools libpthread-stubs0-dev libsm-dev libsm6 libsodi
  libtiff5 libtiff5-dev libtiffxx5 libvpx-dev libvpx5 libwebp6 libx11-dev libx11-doc libxau-dev
  libxpm-dev libxpm4 libxt-dev libxt6 php-common php7.2 php7.2-cli php7.2-common php7.2-gd php7.
```

Step 3: Now, add user with the commands given below.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo adduser nagios
Adding user `nagios' ...
Adding new group `nagios' (1001) ...
Adding new user `nagios' (1001) with group `nagios' ...
Creating home directory `/home/nagios' ...
Copying files from `/etc/skel' ...
```

You can add passwords and enter the user information as shown below.

```
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for nagios
Enter the new value, or press ENTER for the default
        Full Name []: Debashis.intellipaat
        Room Number []: 01
        Work Phone []: 0001
        Home Phone []: 0002
        Other []: no
Is the information correct? [Y/n]
```

Step 4: Run the following commands to complete the user adding process.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo groupadd nagcmd
ubuntu@ip-172-31-32-221:~$ sudo usermod -a -G nagcmd nagios
ubuntu@ip-172-31-32-221:~$ sudo usermod -a -G nagcmd www-data
ubuntu@ip-172-31-32-221:~$
```

Step 5: Now that we are set with the prerequisites, install Nagios Core as shown below.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.2.tar.gz
--2018-12-19 10:38:29--  https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.2.tar.gz
Resolving assets.nagios.com (assets.nagios.com)... 72.14.181.71, 2600:3c00::f03c:91ff:fedf:b821
Connecting to assets.nagios.com (assets.nagios.com)|72.14.181.71|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11301454 (11M) [application/x-gzip]
Saving to: 'nagios-4.4.2.tar.gz'

nagios-4.4.2.tar.gz           100%[==================================================>]  10.78M  11.7MB/s

2018-12-19 10:38:30 (11.7 MB/s) - 'nagios-4.4.2.tar.gz' saved [11301454/11301454]
```

Step 6: Untar the file with the command shown below.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ tar xzf nagios-4.4.2.tar.gz
ubuntu@ip-172-31-32-221:~$
```

Step 7: Enter the Nagios-4.4.2 directory.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~$ cd nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$
```

Step 8: Now with the given command make the required configurations.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo ./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...
```

If the execution ends with the below given setup then we are good to go.

```
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/apache2/sites-available
             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):
```

Now we will make all the configuration work.

ubuntu@ip-172-31-32-221: ~/nagios-4.4.2

```
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo make all
```

If everything is perfect, we should see an output as shown below.

```
*** 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.
Also make sure to include pertinent information that could
help others help you.  This might include:

     - What version of Nagios you are using
     - What version of the plugins you are using
     - Relevant snippets from your config files
     - Relevant error messages from the Nagios log file

For more information on obtaining support for Nagios, visit:

       https://support.nagios.com

************************************************************

Enjoy.
```

Step 9: Run the following command.

ubuntu@ip-172-31-32-221: ~/nagios-4.4.2

```
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo make install
cd ./base && make install
make[1]: Entering directory '/home/ubuntu/nagios-4.4.2/base'
make install-basic
```

After that you will see the output as shown below:

```
*** 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
```

Step 10: Install init and run the following command.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ 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
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$
```

Step 11: Install config and run the following command.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ 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/
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.ci
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-
```

Step 12: Install command mode as shown below:

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ 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 ***

ubuntu@ip-172-31-24-67:~/nagios-4.4.2$
```

Step 13: Before moving ahead run the following commands to copy eventhandlers scripts under the ***libexec*** directory.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo cp -R contrib/eventhandlers/ /usr/local/nagios/libexec/
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo chown -R nagios:nagios /usr/local/nagios/libexec/eventhandlers
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$
```

Step 14: Create Apache configuration

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo nano /etc/apache2/conf-available/nagios.conf
```

Add the below given content to the configuration file.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
  GNU nano 2.9.3                          /etc/apache2/conf-available/nagios.conf

ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin"

<Directory "/usr/local/nagios/sbin">
   Options ExecCGI
   AllowOverride None
   Order allow,deny
   Allow from all
   AuthName "Restricted Area"
   AuthType Basic
   AuthUserFile /usr/local/nagios/etc/htpasswd.users
   Require valid-user
</Directory>

Alias /nagios "/usr/local/nagios/share"

<Directory "/usr/local/nagios/share">
   Options None
   AllowOverride None
   Order allow,deny
   Allow from all
   AuthName "Restricted Area"
   AuthType Basic
   AuthUserFile /usr/local/nagios/etc/htpasswd.users
   Require valid-user
</Directory>
```

Step 15: Add a password as shown below, to complete apache configuration

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
```

Step 16: Enable Apache configuration

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo a2enconf nagios
Enabling conf nagios.
To activate the new configuration, you need to run:
  systemctl reload apache2
```

Step 17: Enable Apache configuration

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo a2enmod cgi rewrite
Module cgi already enabled
Module rewrite already enabled
ubuntu@ip-172-31-18-63:~/nagios-4.4.2$
```

Step 18: Restart apache service.

```
ubuntu@ip-172-31-32-221: ~/nagios-4.4.2
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ sudo service apache2 restart
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$
```

Step 19: Now go to the main directory.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~/nagios-4.4.2$ cd
ubuntu@ip-172-31-32-221:~$
```

Step 20: To install the required Nagios plugin, download the plugins.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ wget http://www.nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz
--2018-12-19 11:04:57--  http://www.nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz
Resolving www.nagios-plugins.org (www.nagios-plugins.org)... 72.14.186.43
Connecting to www.nagios-plugins.org (www.nagios-plugins.org)|72.14.186.43|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz [following]
--2018-12-19 11:04:58--  http://nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz
Resolving nagios-plugins.org (nagios-plugins.org)... 72.14.186.43
Reusing existing connection to www.nagios-plugins.org:80.
HTTP request sent, awaiting response... 200 OK
Length: 2728818 (2.6M) [application/x-gzip]
Saving to: 'nagios-plugins-2.2.1.tar.gz'

nagios-plugins-2.2.1.tar.gz    100%[=============================================>]   2.60M
```

Step 21: Untar the file.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ tar xzf nagios-plugins-2.2.1.tar.gz
ubuntu@ip-172-31-32-221:~$
```

Step 22: Go inside Nagios-2.2.1 directory.

```
ubuntu@ip-172-31-32-221: ~/nagios-plugins-2.2.1
ubuntu@ip-172-31-32-221:~$ cd nagios-plugins-2.2.1
ubuntu@ip-172-31-32-221:~/nagios-plugins-2.2.1$
```

Step 23: Compile the plug-in and then complete the plug-in installation process running the three commands given below:

ubuntu@ip-172-31-32-221: ~/nagios-plugins-2.2.1

```
ubuntu@ip-172-31-32-221:~/nagios-plugins-2.2.1$ sudo ./configure --with-nagios-user=nagios --with
th-openssl
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
```

ubuntu@ip-172-31-32-221: ~/nagios-plugins-2.2.1

```
ubuntu@ip-172-31-32-221:~/nagios-plugins-2.2.1$ sudo make
make  all-recursive
make[1]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1'
Making all in gl
make[2]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1/gl'
rm -f alloca.h-t alloca.h && \
{ echo '/* DO NOT EDIT! GENERATED AUTOMATICALLY! */'; \
  cat ./alloca.in.h; \
} > alloca.h-t && \
mv -f alloca.h-t alloca.h
rm -f c++defs.h-t c++defs.h && \
sed -n -e '/_GL_CXXDEFS/,$p' \
```

ubuntu@ip-172-31-32-221: ~/nagios-plugins-2.2.1

```
ubuntu@ip-172-31-32-221:~/nagios-plugins-2.2.1$ sudo make install
Making install in gl
make[1]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1/gl'
make  install-recursive
make[2]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1/gl'
make[3]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1/gl'
make[4]: Entering directory '/home/ubuntu/nagios-plugins-2.2.1/gl'
if test yes = no; then \
  case 'linux-gnu' in \
    darwin[56]*) \
      need_charset_alias=true ;; \
    darwin* | cygwin* | mingw* | pw32* | cegcc*) \
      need_charset_alias=false ;; \
    *) \
      need_charset_alias=true ;; \
```

Before we can start using Nagios, we going to need to make a small change in the base configurations.

Step 24: Go to the main directory.

ubuntu@ip-172-31-32-221: ~

```
ubuntu@ip-172-31-32-221:~/nagios-plugins-2.2.1$ cd
ubuntu@ip-172-31-32-221:~$
```

Step 25: Get inside ***/usr/local/nagios/etc/nagios.cfg***

```
ubuntu@ip-172-31-32-221: ~
  GNU nano 2.9.3                          /usr/local/nagios/etc/nagios.cfg

cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg

# 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
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
```

Uncomment that line:

```
ubuntu@ip-172-31-32-221: ~
  GNU nano 2.9.3                          /usr/local/nagios/etc/nagios.cfg

cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg

# 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
#cfg_dir=/usr/local/nagios/etc/switches
#cfg_dir=/usr/local/nagios/etc/routers
```

Step 26: Make the following directory.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo mkdir /usr/local/nagios/etc/servers
ubuntu@ip-172-31-32-221:~$
```

Step 27: Verify the configuration before starting Nagios.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

Nagios Core 4.4.2
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2018-08-16
License: GPL

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
```

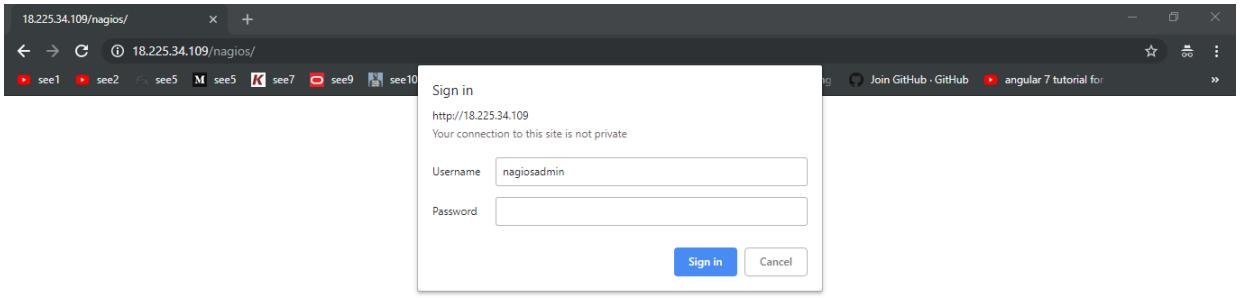
Everything looks fine!

Step 28: Start Nagios

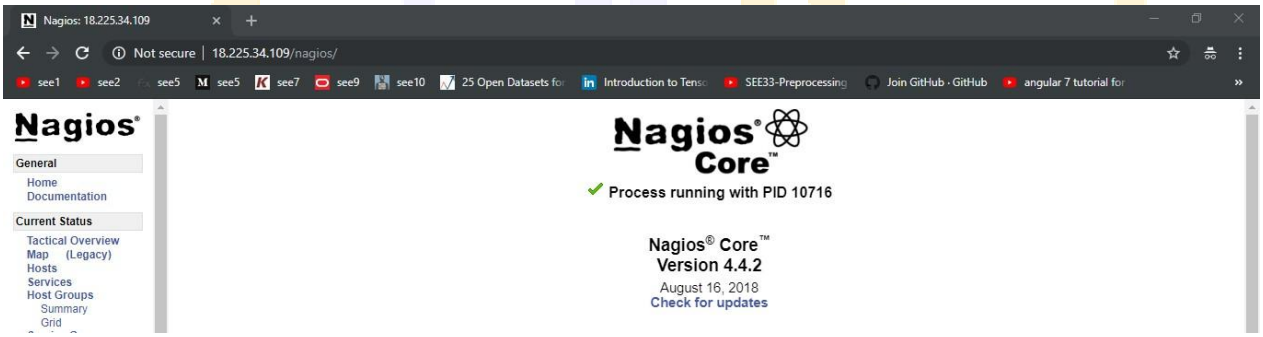
```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo service nagios start
```

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo systemctl enable nagios
ubuntu@ip-172-31-32-221:~$
```

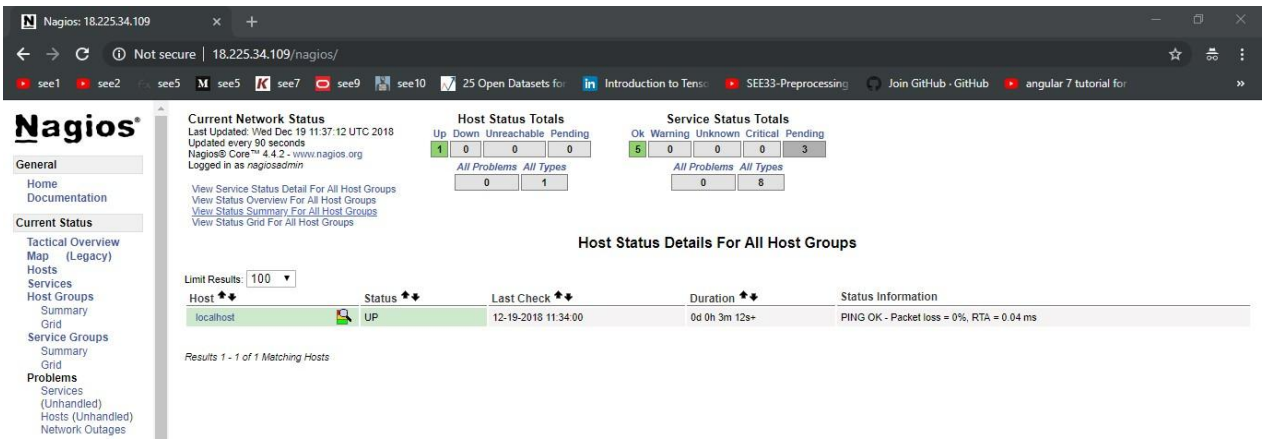
If no error occurs move to the next step on the browser



After entering the user id password, you will land on a page as shown below:



Click on host.



We are all set with the Nagios installation on Master.