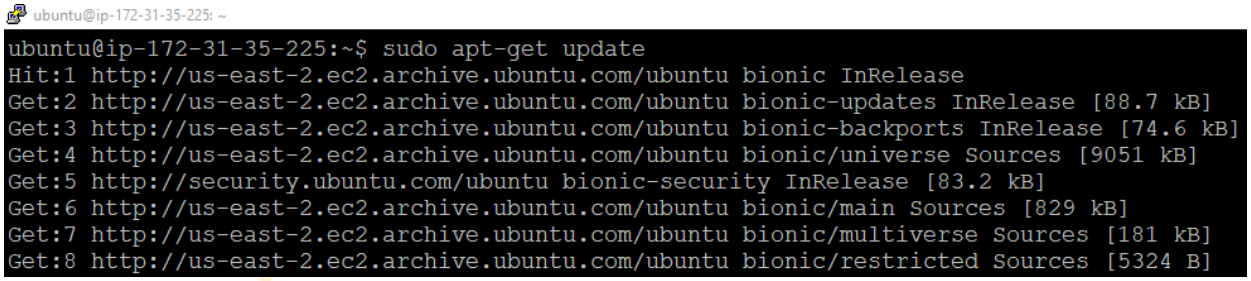
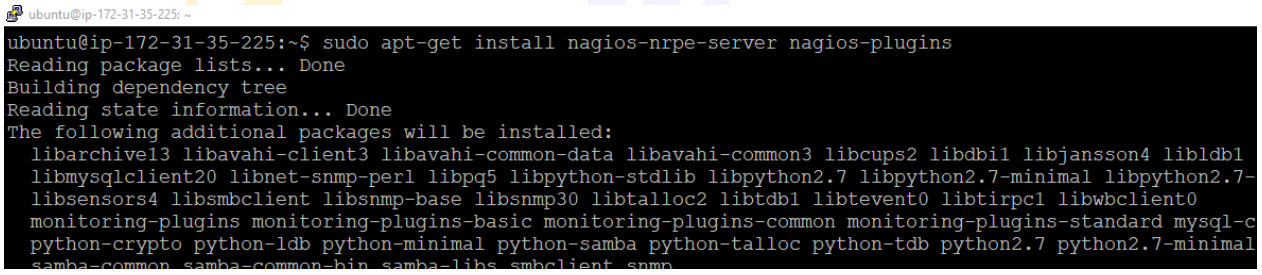
Adding a Host in Nagios

Commands with green font represents master, white font terminal represents slave.
Installing NRPE on slave:

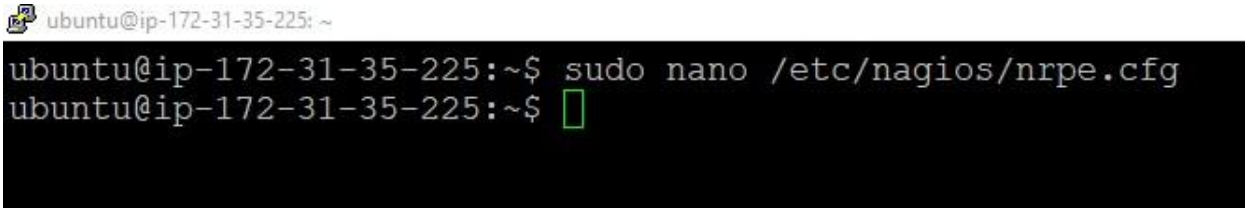
Step 1: On slave run the following command.



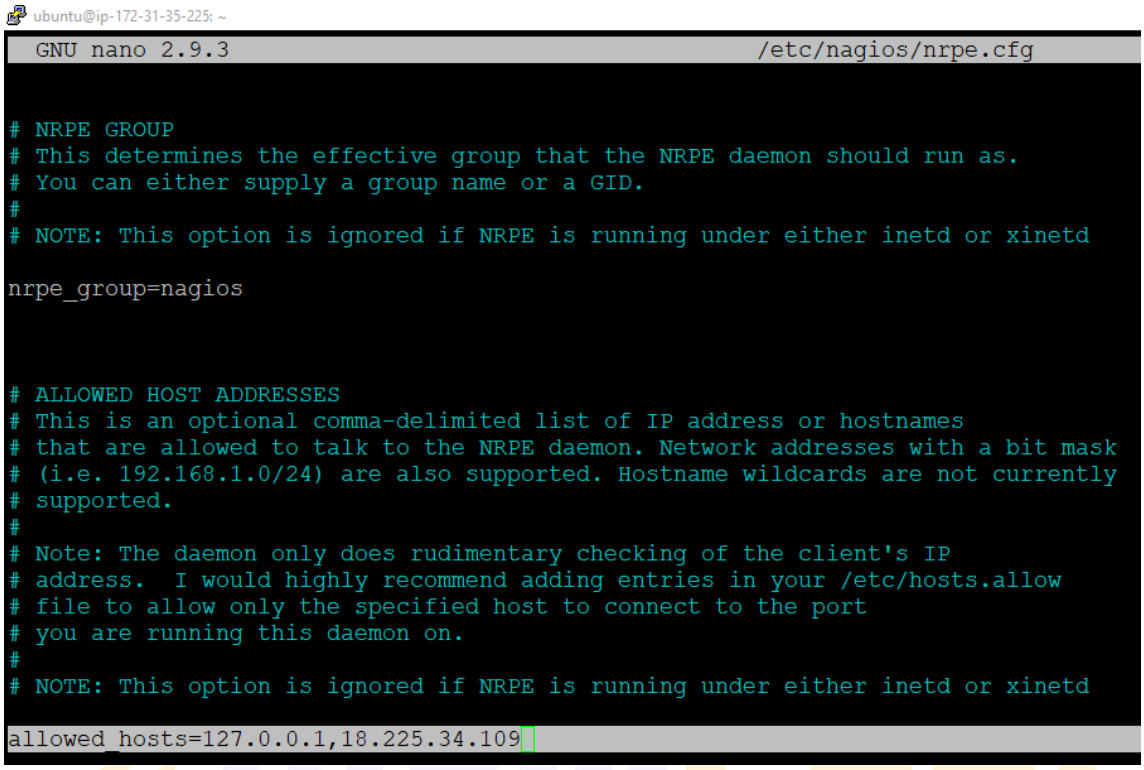
Step 2: Install the required plugins



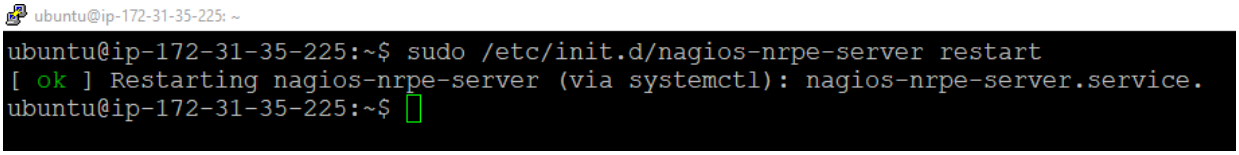
Step 3: Open the configuration file as shown below:



Find the below given line and in place of XXX add your **master IP**
allowed_hosts=127.0.0.1, XXX



Step 4: Start NRPE service as shown below:



Install check_nrpe on Master

Step 1: Now we need to install check_nrpe on Master.

```
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 [88.7 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-updates/main amd64 Packages [461 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [704 kB]
Fetched 1412 kB in 1s (2502 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-32-221:~$
```

Step 2: Install the following package.

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo apt-get install -y autoconf automake gcc libc6 libmcrypt-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
libc6 is already the newest version (2.27-3ubuntu1).
make is already the newest version (4.1-9.1ubuntu1).
make set to manually installed.
gcc is already the newest version (4:7.3.0-3ubuntu2.1).
```

Step 3: Get inside tmp.

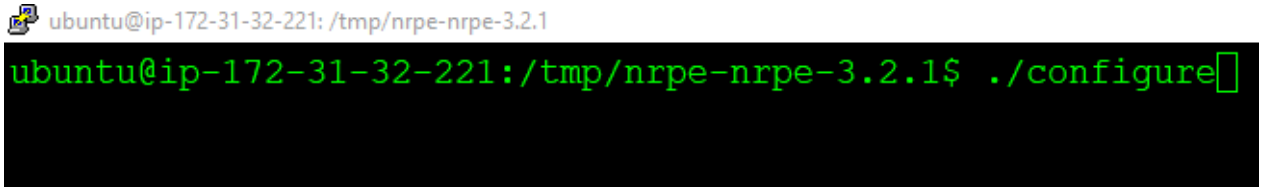
```
ubuntu@ip-172-31-32-221: /tmp
ubuntu@ip-172-31-32-221:~$ cd /tmp
ubuntu@ip-172-31-32-221:/tmp$
```

Step 4: Download the source.

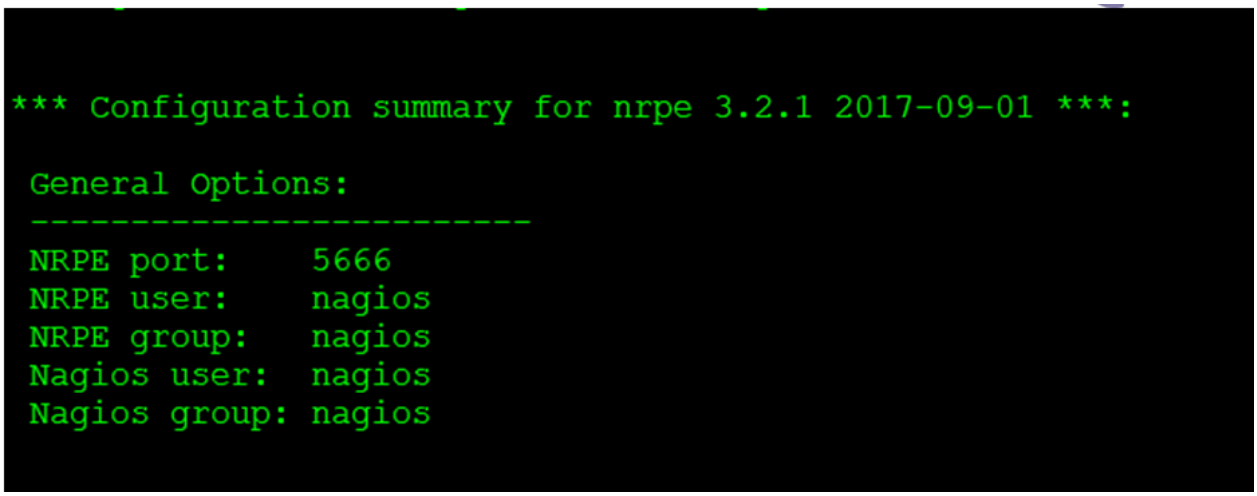
```
ubuntu@ip-172-31-32-221: /tmp
ubuntu@ip-172-31-32-221:/tmp$ wget --no-check-certificate -O nrpe.tar.gz https://github.com/NagiosEnterprises/
/nrpe-3.2.1.tar.gz
--2018-12-19 14:54:06--  https://github.com/NagiosEnterprises/nrpe/archive/nrpe-3.2.1.tar.gz
Resolving github.com (github.com)... 192.30.253.112, 192.30.253.113
Connecting to github.com (github.com)|192.30.253.112|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/NagiosEnterprises/nrpe/tar.gz/nrpe-3.2.1 [following]
--2018-12-19 14:54:06--  https://codeload.github.com/NagiosEnterprises/nrpe/tar.gz/nrpe-3.2.1
Resolving codeload.github.com (codeload.github.com)... 192.30.253.120, 192.30.253.121
Connecting to codeload.github.com (codeload.github.com)|192.30.253.120|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'nrpe.tar.gz'
```

Step 5: Untar the file

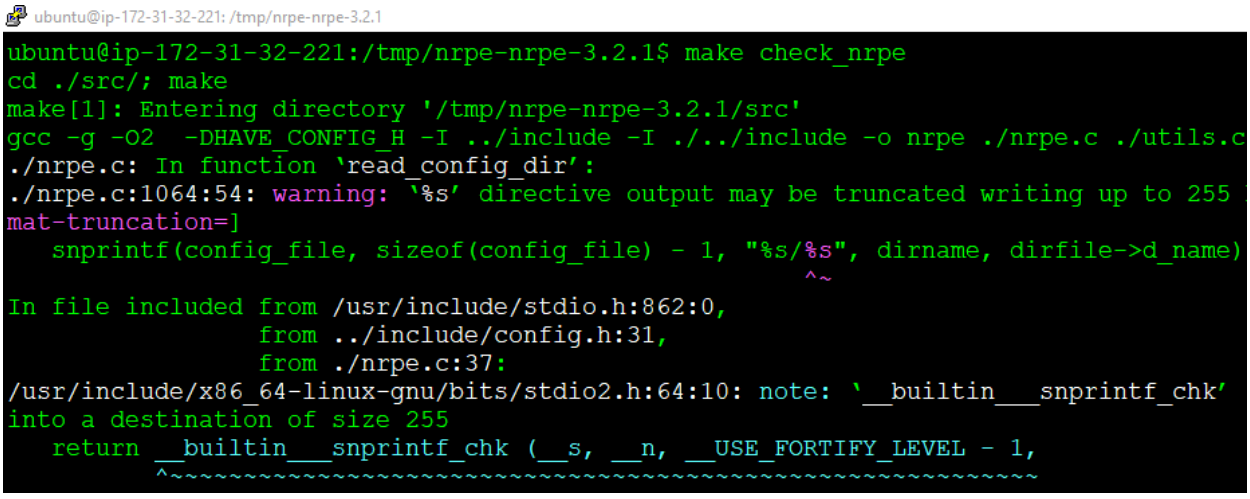
Step 6: Then execute the following command.



It takes some time to configure.



Step 7: Run the check_nrpe command.



Step 8: Now install plugins.

```
ubuntu@ip-172-31-32-221: /tmp/nrpe-nrpe-3.2.1
ubuntu@ip-172-31-32-221:/tmp/nrpe-nrpe-3.2.1$ sudo make install-plugin
cd ./src/; make install-plugin
make[1]: Entering directory '/tmp/nrpe-nrpe-3.2.1/src'
/usr/bin/install -c -m 755 -d /usr/local/nagios/bin
/usr/bin/install -c -m 755 ../uninstall /usr/local/nagios/bin/nrpe-uninstall
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/libexec
/usr/bin/install -c -m 775 -o nagios -g nagios check_nrpe /usr/local/nagios/libexec
make[1]: Leaving directory '/tmp/nrpe-nrpe-3.2.1/src'
ubuntu@ip-172-31-32-221:/tmp/nrpe-nrpe-3.2.1$
```

Step 9: We will check the connection as shown below:

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ /usr/local/nagios/libexec/check_nrpe -H 18.220.227.9
NRPE v3.2.1
ubuntu@ip-172-31-32-221:~$
```

Now we need to make the following configuration file before we observe the connection.

Step 10: Make a configuration file as shown:

```
ubuntu@ip-172-31-32-221: ~
ubuntu@ip-172-31-32-221:~$ sudo nano /usr/local/nagios/etc/servers/Host1.cfg
```

Add the below given content there.
After adding the content, the configuration file should look like this:

```
ubuntu@ip-172-31-27-173: ~
  GNU nano 2.9.3                        /usr/local/nagios/etc/servers/MyHost01.cfg

##############################################################
# Linux Host 001 configuration file
##############################################################

define host {
        use                             Linux-server
        host_name                       Linux_Host_01
        alias                           Linux Host 01
        address                         18.224.140.24
        register                        1
}
define service{
      host_name                         Linux_Host_01
      service_description               PING
      check_command                     check_ping!100.0,20%!500.0,60%
      max_check_attempts                2
      check_interval                    2
      retry_interval                    2
      check_period                      24x7
      check_freshness                   1
      contact_groups                    admins
      notification_interval             2
      notification_period               24x7
      notifications_enabled             1
      register                          1
}
```

Step 11: Now verify the configuration again.

```
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
```

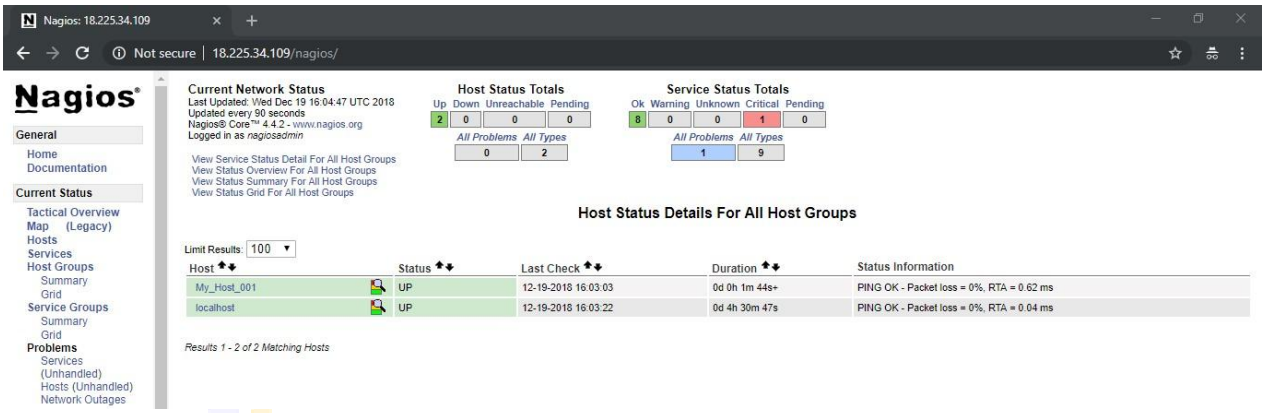
```
Checking objects...
        Checked 9 services.
        Checked 2 hosts.
        Checked 1 host groups.
        Checked 0 service groups.
        Checked 1 contacts.
        Checked 1 contact groups.
        Checked 24 commands.
        Checked 5 time periods.
        Checked 0 host escalations.
        Checked 0 service escalations.
Checking for circular paths...
        Checked 2 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!

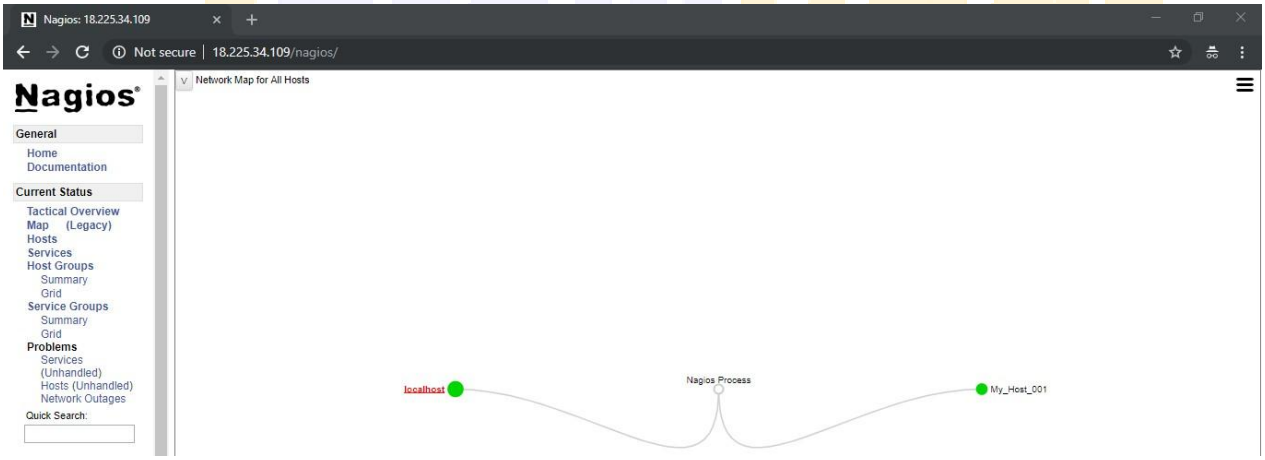
Step 12: Start Nagios.

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

Step 13: Go to the browser. And you can see the slave is added to the connection with a PING service running.



To see the connection map, click on **Map.**



Congratulations! You have successfully setup Nagios Master-Slave Cluster.