

EXPERIMENT - (10)

AIM: To perform Host, service monitoring, windows linux server monitoring using Nagios.

THEORY:-

Nagios is used for continuous monitoring of systems, applications, service and business processes in a DevOps culture.

Important features of Nagios monitoring tool:

Relatively scalable, manageable and secure
Good log and database system.

Informative and attractive web interfaces

Helps you to detect network outages or server crashes.

You can troubleshoot performance issues of servers. If any issues, can be fixed automatically as they are identified during monitoring process.

Utilizes topology to determine dependencies.

Monitor network services like HTTP, SMTP, POP etc.

Helps you to define network host hierarchy using parent tools.

Ability to define event handlers that runs during service or host events for proactive resolution.

Support for implementing redundant monitoring hosts.

Nagios architecture

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

② plugin gets status from remote host.

① Scheduler executes Plugins

Process Scheduler

Nagios

web

Interface

Nagios server

Remote machine

④ Nagios updates

GUI and notifies admin

Plugins

Plugins

③ Plugins send data to Nagios to process.

Scheduler is a component of server part of Nagios. It sends a signal to execute the plugins at remote host.

The plugin gets status from remote host.

The plugin sends data to process scheduler.

The process scheduler updates the GUI and notifications are sent to admins.

CONCLUSION:

Thus we learned about service monitoring using Nagios and successfully monitored a linux server and monitored its different ports and services using Nagios and NRPE.

Steps:

Prerequisites: AWS Free Tier, Nagios Server running on Amazon Linux Machine.

1. To Confirm that Nagios is running **on the server side**, run this *sudo systemctl status nagios* on the “NAGIOS HOST”.

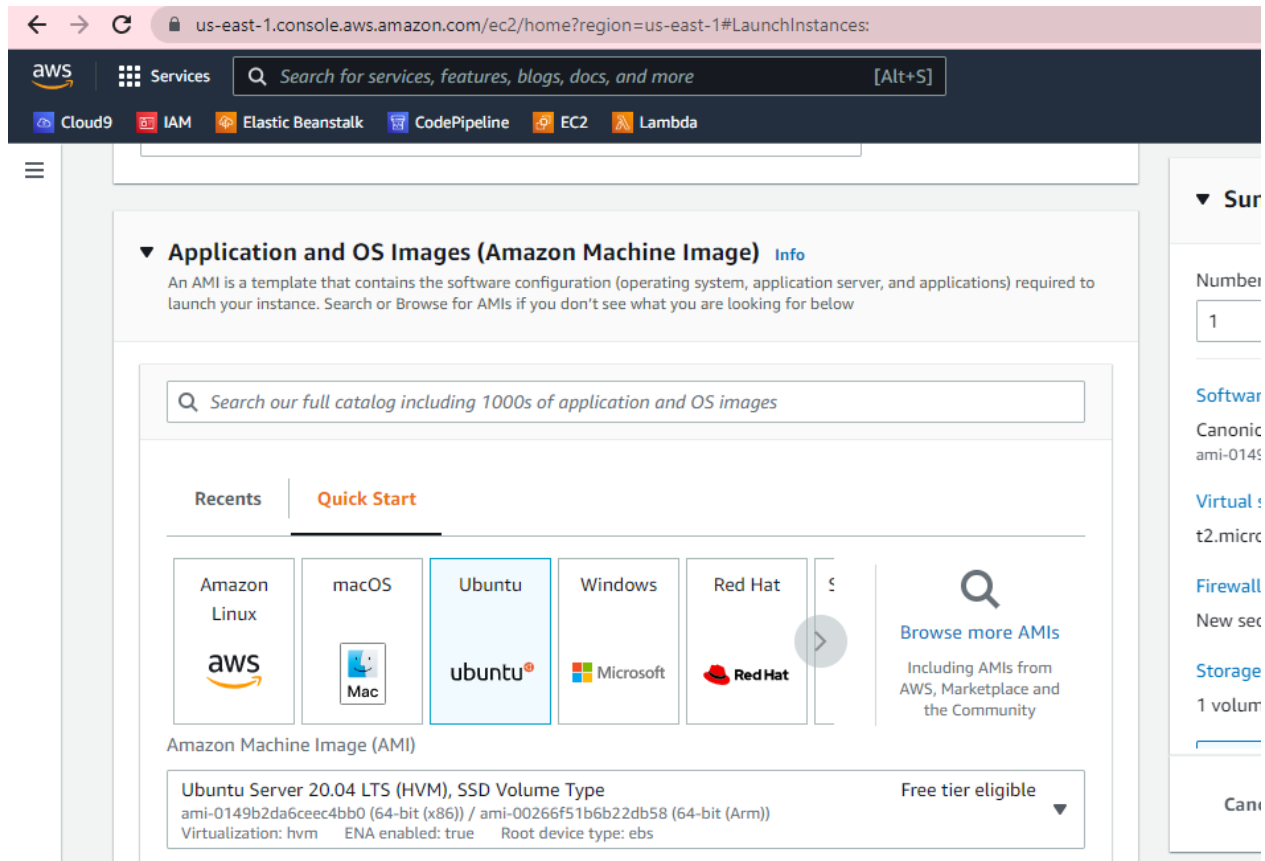
```
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$ sudo systemctl status nagios
● nagios.service - LSB: Starts and stops the Nagios monitoring server
   Loaded: loaded (/etc/rc.d/init.d/nagios; bad; vendor preset: disabled)
   Active: active (running) since Fri 2022-09-23 04:46:16 UTC; 10s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 24411 ExecStart=/etc/rc.d/init.d/nagios start (code=exited, status=0/SUCCESS)
    CGroup: /system.slice/nagios.service
            └─24432 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
               24434 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               24435 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               24436 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               24437 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
               24438 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Sep 23 04:46:16 ip-172-31-21-133.ec2.internal nagios[24432]: nerd: Channel hostchecks registered successfully

i-01930fac31f9a3f9b (nagios-host)
```

You can proceed if you get this message.

2. Before we begin,
To monitor a Linux machine, create an **Ubuntu 20.04 server EC2** Instance in AWS.



Provide it with the same security group as the Nagios Host and name it **'linux-client'** alongside the host.

For now, leave this machine as is, and go back to your nagios HOST machine.

Find instance by attribute or tag (case-sensitive)						
	Name	Instance ID	Instance state	Instance type	Status check	
<input type="checkbox"/>	Myebs-env	i-0e657a0c7aa54bf59	Running	t2.micro	2/2 checks passed	
<input type="checkbox"/>	nagios-host	i-01930fac31f9a3f9b	Running	t2.micro	2/2 checks passed	
<input type="checkbox"/>	linux-client	i-0626cf3ac484dac8e	Terminated	t2.micro	-	
<input checked="" type="checkbox"/>	linux-host	i-08b31d77a0fe2f2eb	Pending	t2.micro	-	

Instance: i-08b31d77a0fe2f2eb (linux-host)

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
---------	----------	------------	---------	---------------	------------	------

3. On the server, run this command

```
ps -ef | grep nagios
```



```
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$ ps -ef | grep nagios
nagios 24432 1 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
nagios 24434 24432 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 24435 24432 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 24436 24432 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 24437 24432 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
nagios 24438 24432 0 04:46 ? 00:00:00 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
ec2-user 24539 27314 0 04:56 pts/0 00:00:00 grep --color=auto nagios
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$
```

4. Become a root user and create 2 folders

```
sudo su
mkdir /usr/local/nagios/etc/objects/monitorhosts
mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
```

```
[ec2-user@ip-172-31-21-133 nagios-plugins-2.0.3]$ sudo su
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# mkdir /usr/local/nagios/etc/objects/monitorhosts/linuxhosts
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#
```

5. Copy the sample localhost.cfg file to linuxhost folder

```
cp /usr/local/nagios/etc/objects/localhost.cfg
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

```
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# cp /usr/local/nagios/etc/objects/localhost.cfg /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#
```

6. Open linuxserver.cfg using nano and make the following changes

```
nano
/usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
```

Change the hostname to linuxserver (EVERYWHERE ON THE FILE)

```
GNU nano 2.9.8 /usr/local/nagios/etc/objects/monitorhosts/linuxhosts/linuxserver.cfg
#####
#####
# Define a host for the local machine
define host{
    use                linux-server          ; Name of host template to use
                                           ; This host definition will inherit all variables that are defined
                                           ; in (or inherited by) the linux-server host template definition.

    host_name          linuxserver
    alias              linuxserver
    address            127.0.0.1
}

#####
#####
#
# HOST GROUP DEFINITION

^G Get Help  ^O Write Out  ^W Where Is  ^C Cut Text   ^J Justify    ^_ Cur Pos   ^U Undo      ^M Mark Text  ^- To Bracket
^X Exit      ^R Read File  ^\ Replace   ^_ Uncut Text ^T To Spell   ^_ Go To Line ^E Redo      ^- Copy Text  ^- WhereIs Next
```

Change address to the public IP address of your **LINUX CLIENT**.

```
#####
# Define a host for the local machine
define host{
    use                linux-server          ; Name of host template to use
                                           ; This host definition will inherit all variables that are defined
                                           ; in (or inherited by) the linux-server host template definition.
    host_name          linuxserver
    alias              linuxserver
    address            107.23.206.182
}

#####
```

Change hostgroup_name under hostgroup to linux-servers1

Everywhere else on the file, change the hostname to linuxserver instead of localhost.

```
#####
# Define an optional hostgroup for Linux machines
define hostgroup{
    hostgroup_name     linux-servers1 ; The name of the hostgroup
    alias              Linux Servers ; Long name of the group
    members            linuxserver    ; Comma separated list of hosts that belong to this group
}

#####
```

7. Open the Nagios Config file and add the following line

nano /usr/local/nagios/etc/nagios.cfg

##Add this line

cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

```
GNU nano 2.9.8 /usr/local/nagios/etc/nagios.cfg
#####
# NAGIOS.CFG - Sample Main Config File for Nagios 4.0.8
# Read the documentation for more information on this configuration
# file. I've provided some comments here, but things may not be so
# clear without further explanation.
#
#####
# LOG FILE
# This is the main log file where service and host events are logged
# for historical purposes. This should be the first option specified
# in the config file
cfg_dir=/usr/local/nagios/etc/objects/monitorhosts/

# OBJECT CONFIGURATION FILE(S)

[ Read 1334 lines ]
^G Get Help      ^C Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo        M-A Mark Text    M-
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line    M-E Redo        M-6 Copy Text  M-
```

8. Verify the configuration files

sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

```

Checking objects...
  Checked 16 services.
  Checked 2 hosts.
  Checked 2 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

Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#

```

```

i-01930fac31f9a3f9b (nagios-host)
PublicIPs: 54.160.162.142 PrivateIPs: 172.31.21.133

```

You are good to go if there are no errors.

9. Restart the nagios service

```
service nagios restart
```

```

Things look okay - No serious problems were detected during the pre-flight check
[root@ip-172-31-21-133 nagios-plugins-2.0.3]# service nagios restart
Restarting nagios (via systemctl): [ OK ]
[root@ip-172-31-21-133 nagios-plugins-2.0.3]#

```

```
i-01930fac31f9a3f9b (nagios-host)
```

Now it is time to switch to the client machine.

10. SSH into the machine or simply use the EC2 Instance Connect

feature.

11. Make a package index update and install gcc, nagios-nrpe-server and the plugins.

```

sudo apt update -y
sudo apt install gcc -y
sudo apt install -y nagios-nrpe-server nagios-plugins

```



```

Creating config file /etc/nagios-plugins/config/snmp.cfg with new version
Setting up monitoring-plugins (2.2-6ubuntu1.2) ...
Setting up python3-ldb (2:2.2.3-0ubuntu0.20.04.3) ...
Setting up libavahi-client3:amd64 (0.7-4ubuntu7.1) ...
Setting up libcups2:amd64 (2.3.1-9ubuntu1.2) ...
Setting up samba-lsmb:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up libsmclient:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up smbclient (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up samba-dsdb-modules:amd64 (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up python3-samba (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Setting up samba-common-bin (2:4.13.17~dfsg-0ubuntu1.20.04.1) ...
Checking smb.conf with testparm
Load smb config files from /etc/samba/smb.conf
Loaded services file OK.
Weak crypto is allowed
Server role: ROLE_STANDALONE

Done
Processing triggers for systemd (245.4-4ubuntu3.17) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
ubuntu@ip-172-31-94-97:~$

```

i-08b31d77a0fe2f2eb (linux-host)

PublicIPs: 107.23.206.182 PrivateIPs: 172.31.94.97

12. Open nrpe.cfg file to make changes.

```
sudo nano /etc/nagios/nrpe.cfg
```

Under `allowed_hosts`, add your nagios host IP address like so

```

GNU nano 4.8 /etc/nagios/nrpe.cfg
# 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.
#
# NOTE: This option is ignored if NRPE is running under either inetd or xinetd

allowed_hosts=127.0.0.1,::,54.160.162.142

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow clients
# to specify arguments to commands that are executed. This option only works
# if the daemon was configured with the --enable-command-args configure script
# option.
#
^G Get Help      ^C Write Out    ^W Where Is     ^R Cut Text     ^J Justify      ^C Cur Pos      M-U Undo
^X Exit          ^R Read File    ^\ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line     M-E Redo

```

13. Restart the NRPE server

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

14. Now, check your nagios dashboard and you'll see a new host being added.

Click on Hosts.

← → ↻ Not secure | 54.160.162.142/nagios/

Nagios®

General
Home
Documentation

Current Status
Tactical Overview
Map
Hosts
Services
Host Groups
Summary
Grid
Service Groups
Summary
Grid
Problems
Services
(Unhandled)
Hosts (Unhandled)
Network Outages
Quick Search:
Reports

Current Network Status
Last Updated: Fri Sep 23 05:18:18 UTC 2022
Updated every 90 seconds
Nagios® Core™ 4.0.8 - www.nagios.org
Logged in as nagiosadmin

Host Status Totals

Up	Down	Unreachable	Pending
2	0	0	0

All Problems: 0 All Types: 2

Service Status Totals

Ok	Warning	Unknown	Critical	Pending
12	1	0	3	0

All Problems: 4 All Types: 16

View Service Status Detail For All Host Groups
View Status Overview For All Host Groups
View Status Summary For All Host Groups
View Status Grid For All Host Groups

Host Status Details For All Host Groups

Limit Results: 100

Host	Status	Last Check	Duration	Status Information
linuxserver	UP	09-23-2022 05:17:20	0d 0h 5m 43s	PING OK - Packet loss = 0%, RTA = 0.97 ms
localhost	UP	09-23-2022 05:16:08	0d 0h 31m 25s	PING OK - Packet loss = 0%, RTA = 0.05 ms

Results 1 - 2 of 2 Matching Hosts

Click on linuxserver to see the host details

← → ↻ Not secure | 54.160.162.142/nagios/

Nagios®

General
Home
Documentation

Current Status
Tactical Overview
Map
Hosts
Services
Host Groups
Summary
Grid
Service Groups
Summary
Grid
Problems
Services
(Unhandled)
Hosts (Unhandled)
Network Outages
Quick Search:
Reports

Host Information
Last Updated: Fri Sep 23 05:18:40 UTC 2022
Updated every 90 seconds
Nagios® Core™ 4.0.8 - www.nagios.org
Logged in as nagiosadmin

View Status Detail For This Host
View Alert History For This Host
View Trends For This Host
View Alert Histogram For This Host
View Availability Report For This Host
View Notifications For This Host

Host: **linuxserver (linuxserver)**
Member of: linux-servers1
107.23.206.182

Host State Information

Host Status: **UP** (for 0d 0h 6m 5s)
Status Information: PING OK - Packet loss = 0%, RTA = 0.83 ms
Performance Data: rta=0.834000ms;3000.000000;5000.000000;0.000000 pi=0%;80;100;0
Current Attempt: 1/10 (HARD state)
Last Check Time: 09-23-2022 05:18:20
Check Type: ACTIVE
Check Latency / Duration: 0.000 / 4.092 seconds
Next Scheduled Active Check: 09-23-2022 05:23:24
Last State Change: 09-23-2022 05:12:35
Last Notification: N/A (notification 0)
Is This Host Flapping? **NO** (0.00% state change)
In Scheduled Downtime? **NO**
Last Update: 09-23-2022 05:18:37 (0d 0h 0m 3s 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

Host Comments

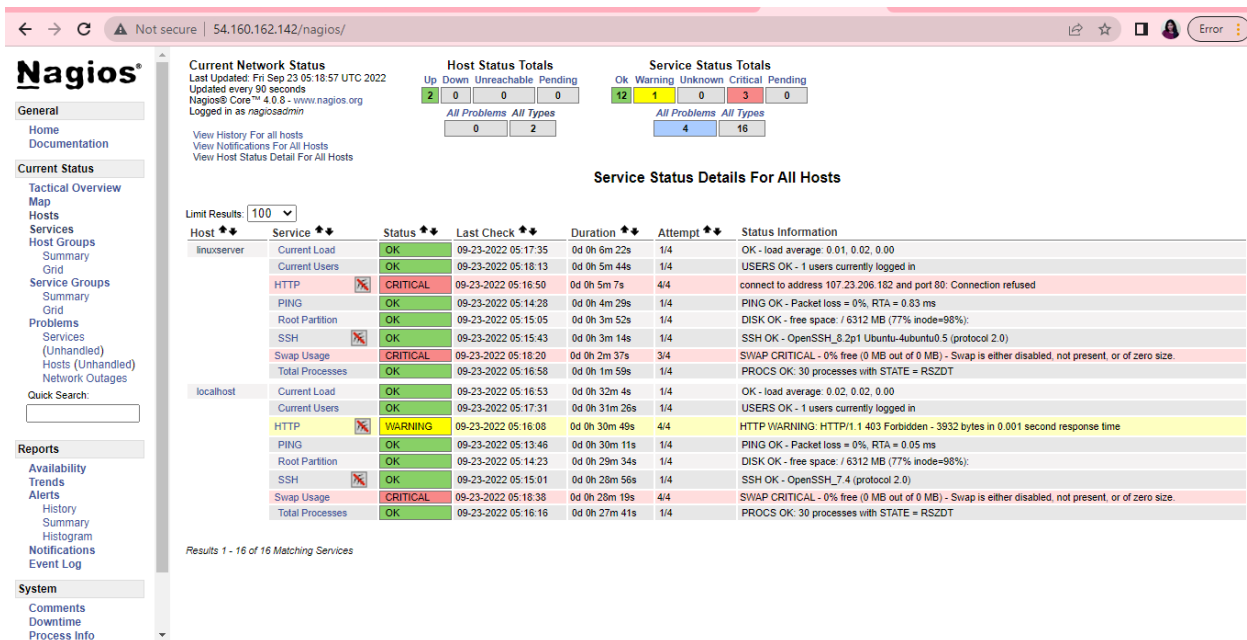
Add a new comment Delete all comments

Entry Time	Author	Comment	Comment ID	Persistent	Type	Expires	Actions
This host has no comments associated with it							

Availability
Trends
Alerts
History
Summary
Histogram
Notifications
Event Log

System
Comments
Downtime
Process Info

You can click Services to see all services and ports being monitored.



As you can see, we have our linuxserver up and running. It is showing critical status on HTTP due to permission errors and swap because there is no partition created.

In this case, we have monitored -

Servers: 1 linux server

Services: swap

Ports: 22, 80 (ssh, http)

Processes: User status, Current load, total processes, root partition, etc.

Recommended Cleanup

- Terminate both of your EC-2 instances to avoid charges.
- Delete the security group if you created a new one (it won't affect your bill, you may avoid it)

Conclusion:

Thus, we learned about service monitoring using Nagios and successfully monitored a Linux Server and monitored its different ports and services using Nagios and NRPE.