Experiment no . 10

Aim : To perform Port, Service monitoring, Windows/Linux server monitoring using Nagios.

Theory :

Linux – Network Monitoring Tools

Network monitoring is using a system (hardware or software) that continuously observes your network and the data flows through it, depending on how the monitoring solution actually functions and informs the network administrator. We can keep a check on all the activities of our network easily. While Network management we need network Monitoring.

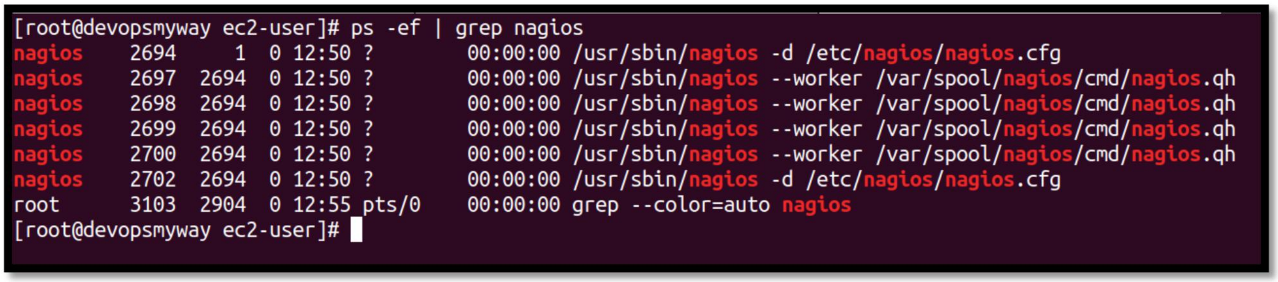
To monitor Windows Machines you will need to follow several steps and they are:

Install NSClient++ addon on the Windows Machine.
Configure Nagios Server for monitoring Windows Machine.
Add new host and service definitions for Windows machine monitoring.
Restart the Nagios Service.
To make this guide simple and easier, a few of configuration already done for you in the Nagios installation.

A check_nt command definition already added to the command.cfg file. This definition command is used by check_nt plugin to monitor Windows services.
A windows-server host template already created in the templates.cfg file. This template allows you to add new Windows host definitions.

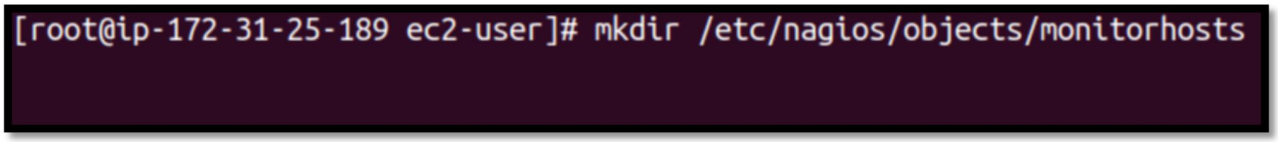
Check Nagios Configuration path
Login to Nagios Server. Use the following command to check the Nagios configuration path.

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



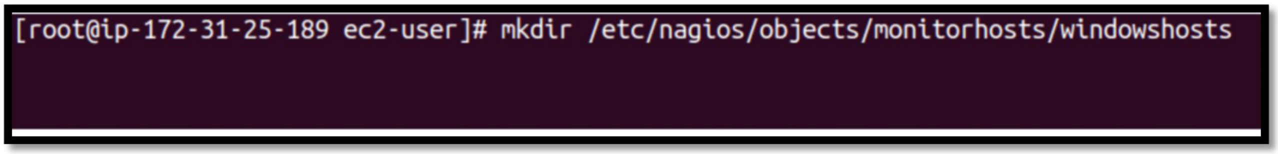
```
[root@devopsmyway ec2-user]# ps -ef | grep nagios
nagios    2694     1  0 12:50 ?        00:00:00 /usr/sbin/nagios -d /etc/nagios/nagios.cfg
nagios    2697  2694  0 12:50 ?        00:00:00 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
nagios    2698  2694  0 12:50 ?        00:00:00 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
nagios    2699  2694  0 12:50 ?        00:00:00 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
nagios    2700  2694  0 12:50 ?        00:00:00 /usr/sbin/nagios --worker /var/spool/nagios/cmd/nagios.qh
nagios    2702  2694  0 12:50 ?        00:00:00 /usr/sbin/nagios -d /etc/nagios/nagios.cfg
root      3103  2904  0 12:55 pts/0    00:00:00 grep --color=auto nagios
[root@devopsmyway ec2-user]# 
```

Create config files for Windows and Linux host
Create a directory, say montitorhosts in /etc/nagios/objects/
$ mkdir /etc/nagios/objects/monitorhosts



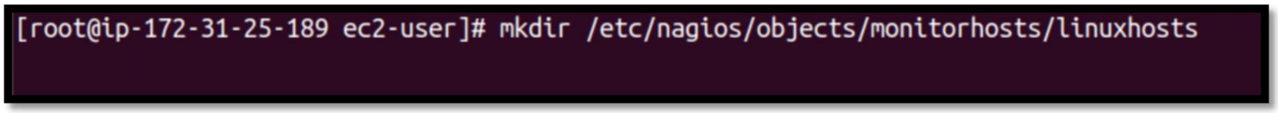
```
[root@ip-172-31-25-189 ec2-user]# mkdir /etc/nagios/objects/monitorhosts
```

Create two directories, say linuxhosts and windowshosts in/etc/nagios/objects/monitorhosts/
$ mkdir /etc/nagios/objects/monitorhosts/windowshosts



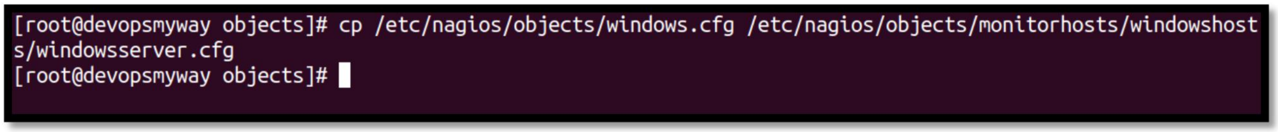
```
[root@ip-172-31-25-189 ec2-user]# mkdir /etc/nagios/objects/monitorhosts/windowshosts
```

$ mkdir /etc/nagios/objects/monitorhosts/linuxhosts



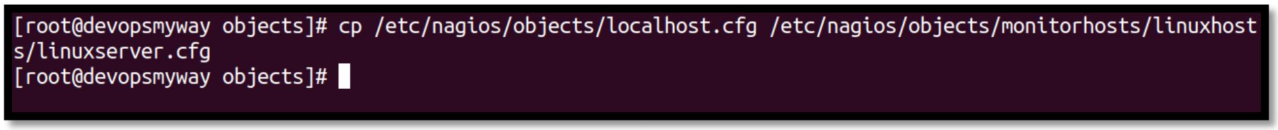
```
[root@ip-172-31-25-189 ec2-user]# mkdir /etc/nagios/objects/monitorhosts/linuxhosts
```

$ cp /etc/nagios/objects/windows.cfg /etc/nagios/objects/monitorhosts/windowshosts/windowsserver.cfg



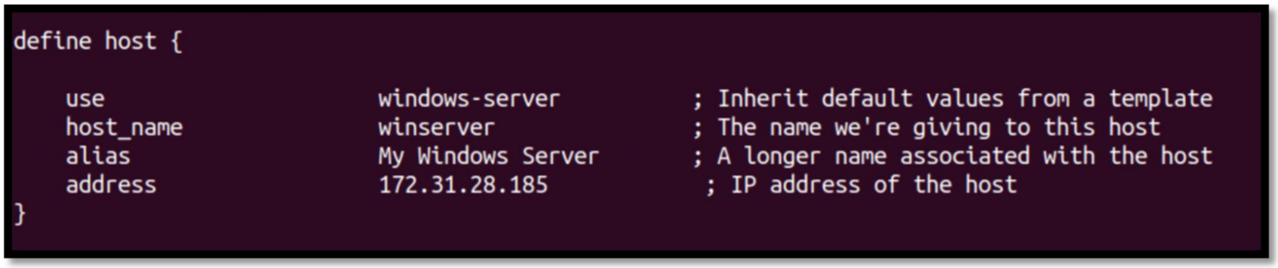
```
[root@devopsmyway objects]# cp /etc/nagios/objects/windows.cfg /etc/nagios/objects/monitorhosts/windowshost
s/windowsserver.cfg
[root@devopsmyway objects]# 
```

$ cp /etc/nagios/objects/localhost.cfg /etc/nagios/objects/monitorhosts/linuxhosts/linuxserver.cfg

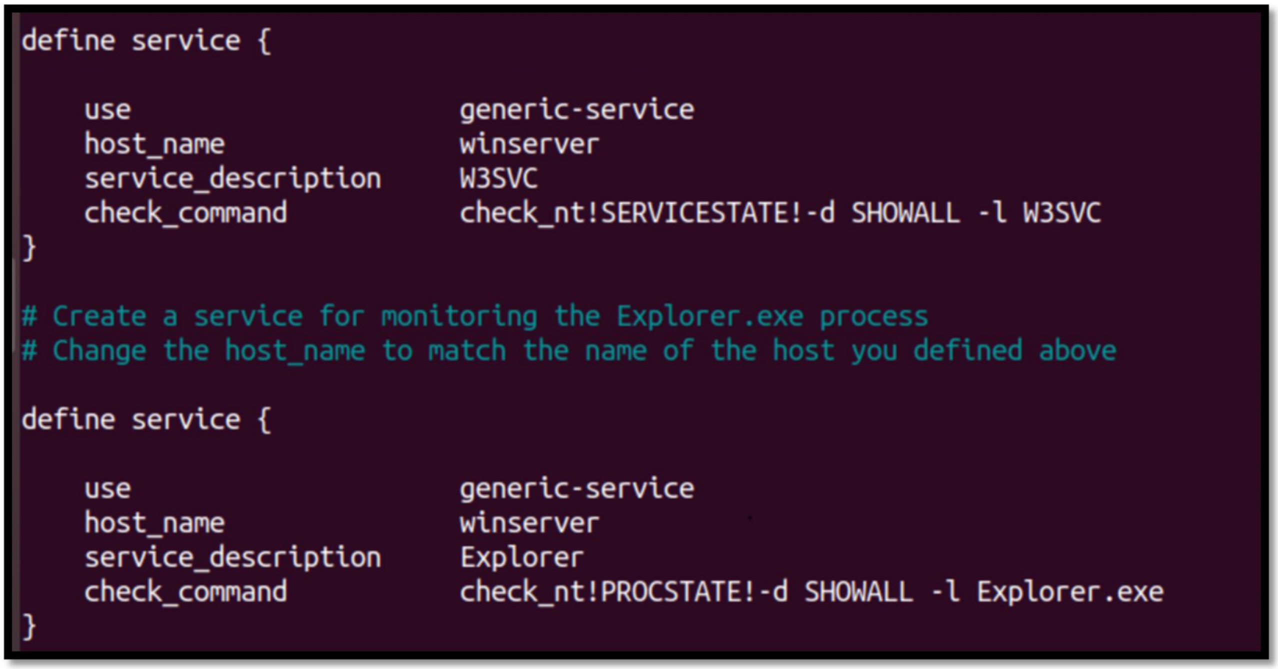


```
[root@devopsmyway objects]# cp /etc/nagios/objects/localhost.cfg /etc/nagios/objects/monitorhosts/linuxhost
s/linuxserver.cfg
[root@devopsmyway objects]#
```

$ nano /etc/nagios/objects/monitorhosts/windowshosts/windowsserver.cfg

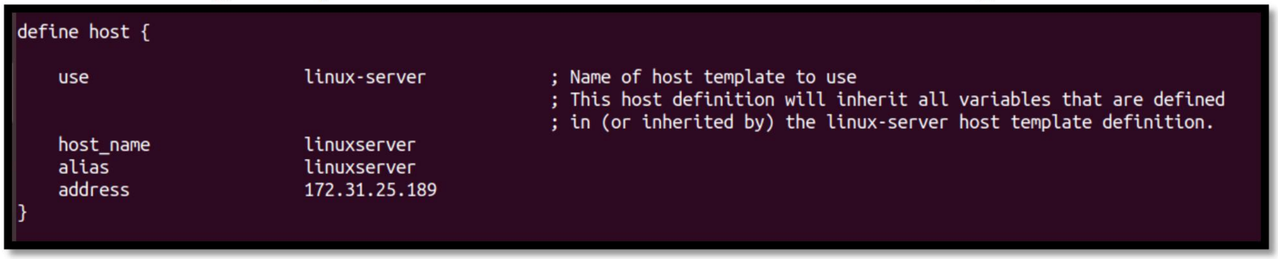


```
define host {

    use                     windows-server          ; Inherit default values from a template
    host_name               winserver               ; The name we're giving to this host
    alias                   My Windows Server       ; A longer name associated with the host
    address                 172.31.28.185            ; IP address of the host
}
```

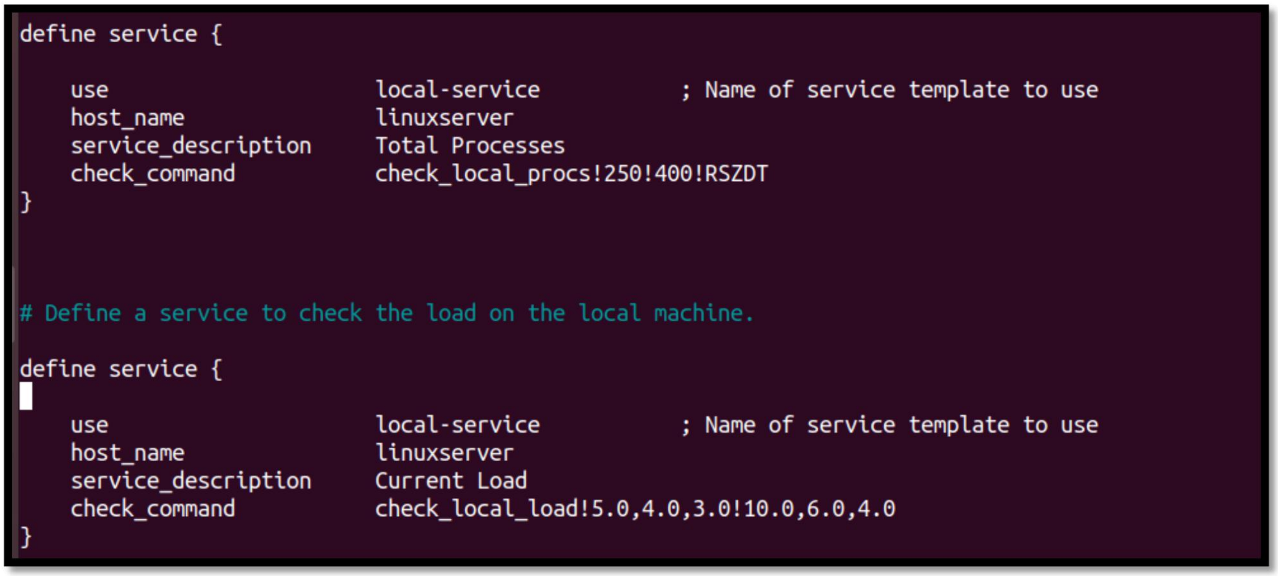


```
define service {

    use                     generic-service
    host_name               winserver
    service_description     W3SVC
    check_command           check_nt!SERVICESTATE!-d SHOWALL -l W3SVC
}

# Create a service for monitoring the Explorer.exe process
# Change the host_name to match the name of the host you defined above

define service {

    use                     generic-service
    host_name               winserver
    service_description     Explorer
    check_command           check_nt!PROCSTATE!-d SHOWALL -l Explorer.exe
}
```

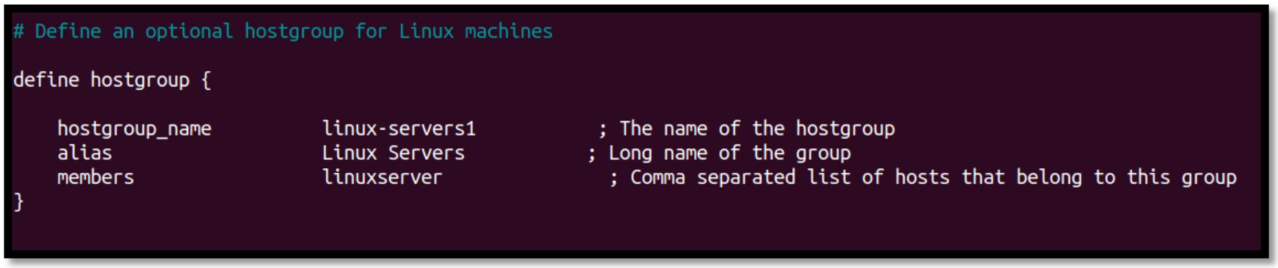
$ nano /etc/nagios/objects/monitorhosts/linuxhosts/linuxserver.cfg



```
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                 172.31.25.189
}
```



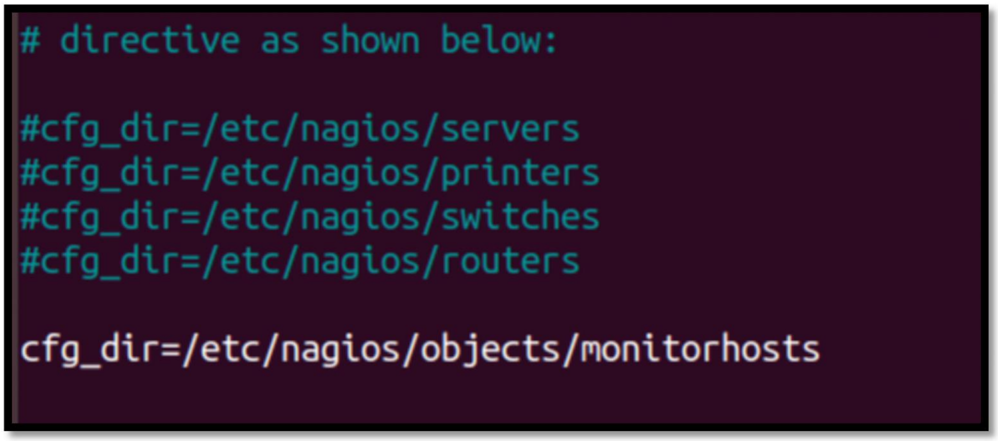
```
define service {

    use                     local-service           ; Name of service template to use
    host_name               linuxserver
    service_description     Total Processes
    check_command           check_local_procs!250!400!RSZDT
}



# Define a service to check the load on the local machine.

define service {

    use                     local-service           ; Name of service template to use
    host_name               linuxserver
    service_description     Current Load
    check_command           check_local_load!5.0,4.0,3.0!10.0,6.0,4.0
}
```



```
# 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
}
```

$ nano /etc/nagios/nagios.cfg

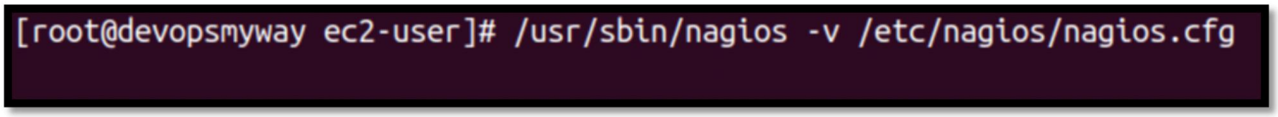
cfg_dir=/etc/nagios/objects/monitorhosts



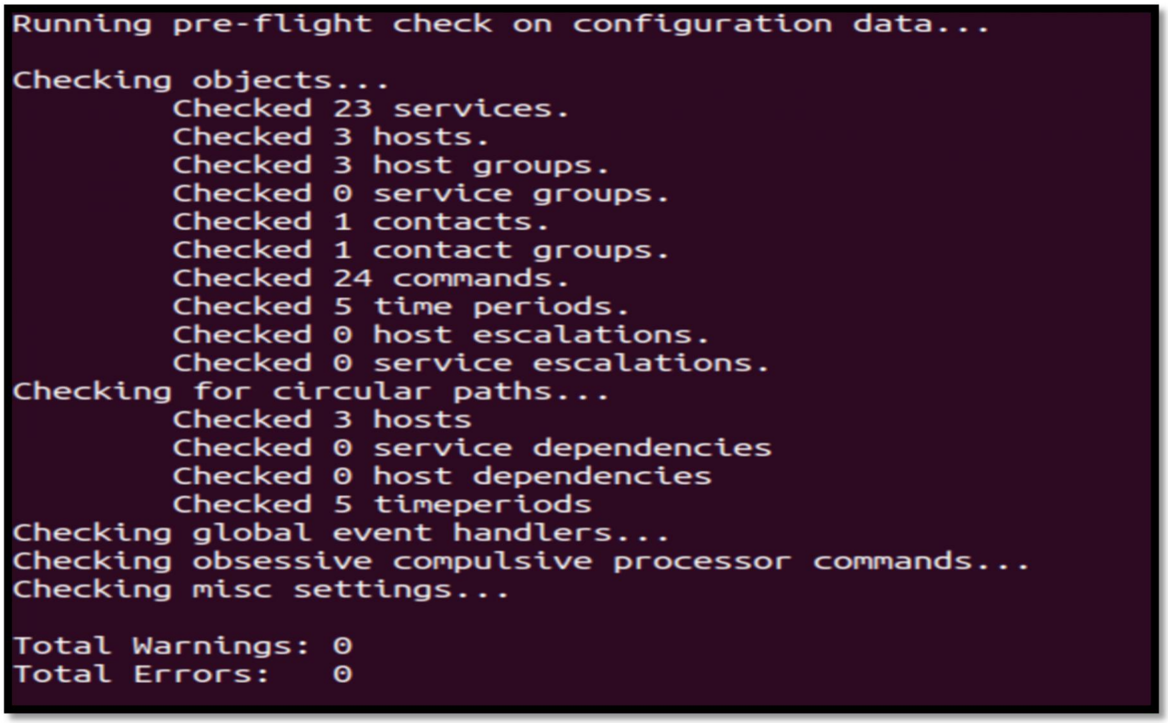
```
# directive as shown below:

#cfg_dir=/etc/nagios/servers
#cfg_dir=/etc/nagios/printers
#cfg_dir=/etc/nagios/switches
#cfg_dir=/etc/nagios/routers

cfg_dir=/etc/nagios/objects/monitorhosts
```

Check the Nagios Configuration

$ /usr/sbin/nagios -v /etc/nagios/nagios.cfg



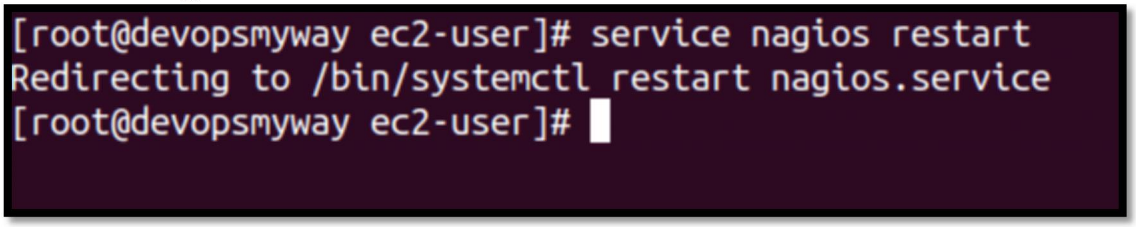
```
[root@devopsmyway ec2-user]# /usr/sbin/nagios -v /etc/nagios/nagios.cfg
```



```
Running pre-flight check on configuration data...

Checking objects...
        Checked 23 services.
        Checked 3 hosts.
        Checked 3 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 3 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
```

Restart Nagios Service

$ service nagios restart

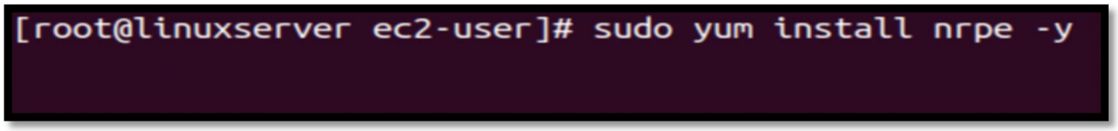


```
[root@devopsmyway ec2-user]# service nagios restart
Redirecting to /bin/systemctl restart nagios.service
[root@devopsmyway ec2-user]# 
```

Configuration in Linux host

Login to Linux Server and Install nrpe plugin.

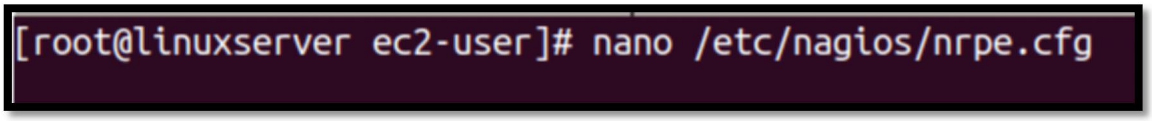
$ sudo yum install nrpe -y



```
[root@linuxserver ec2-user]# sudo yum install nrpe -y
```

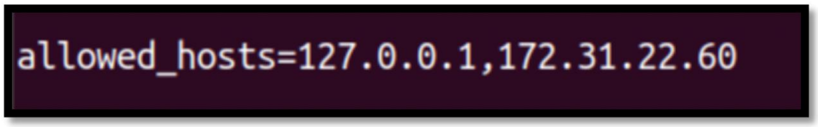
Open nrpe config file

$ nano /etc/nagios/nrpe.cfg



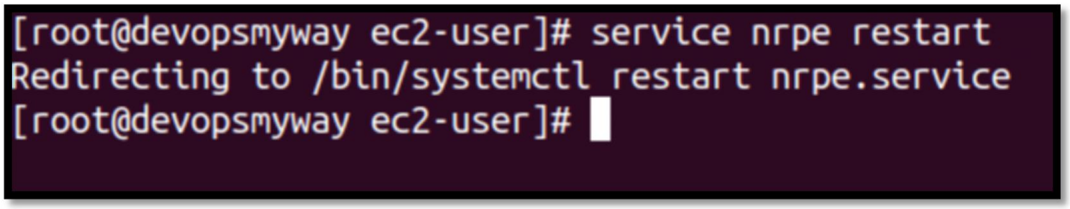
```
[root@linuxserver ec2-user]# nano /etc/nagios/nrpe.cfg
```

Put the IP address of Nagios Server in allowed_hosts in nrpe.cfg.



```
allowed_hosts=127.0.0.1,172.31.22.60
```

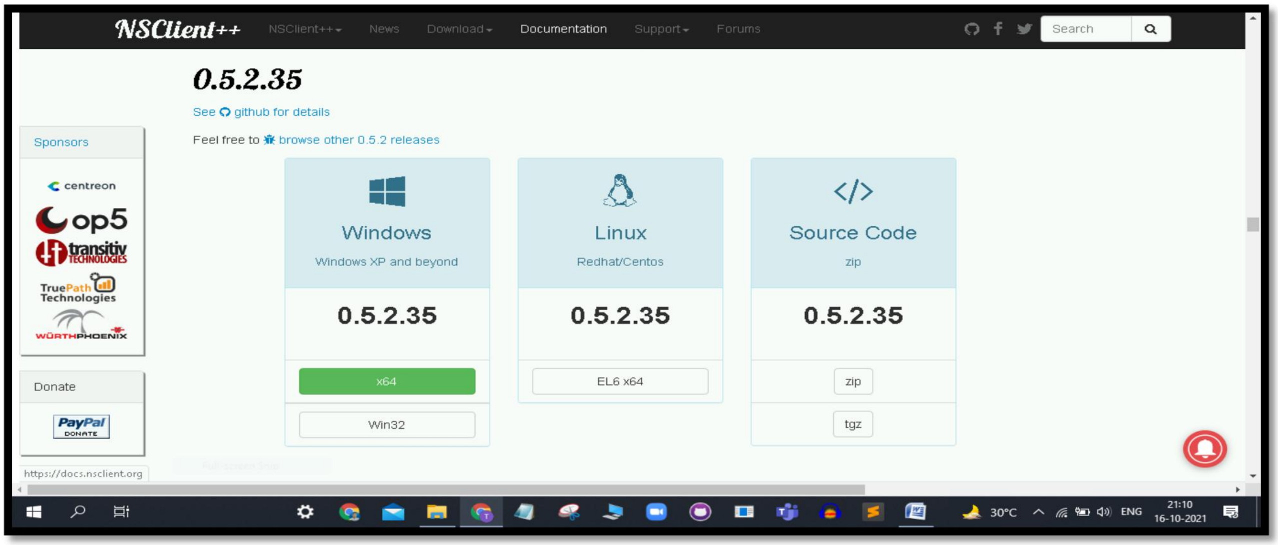
Restart nrpe service



```
[root@devopsmyway ec2-user]# service nrpe restart
Redirecting to /bin/systemctl restart nrpe.service
[root@devopsmyway ec2-user]# 
```

Configuration in Windows host
Log in to your Windows Server and download nsclient++ and install it. You can use the following link to download the nsclient++ for windows.

http://nsclient.org/download/

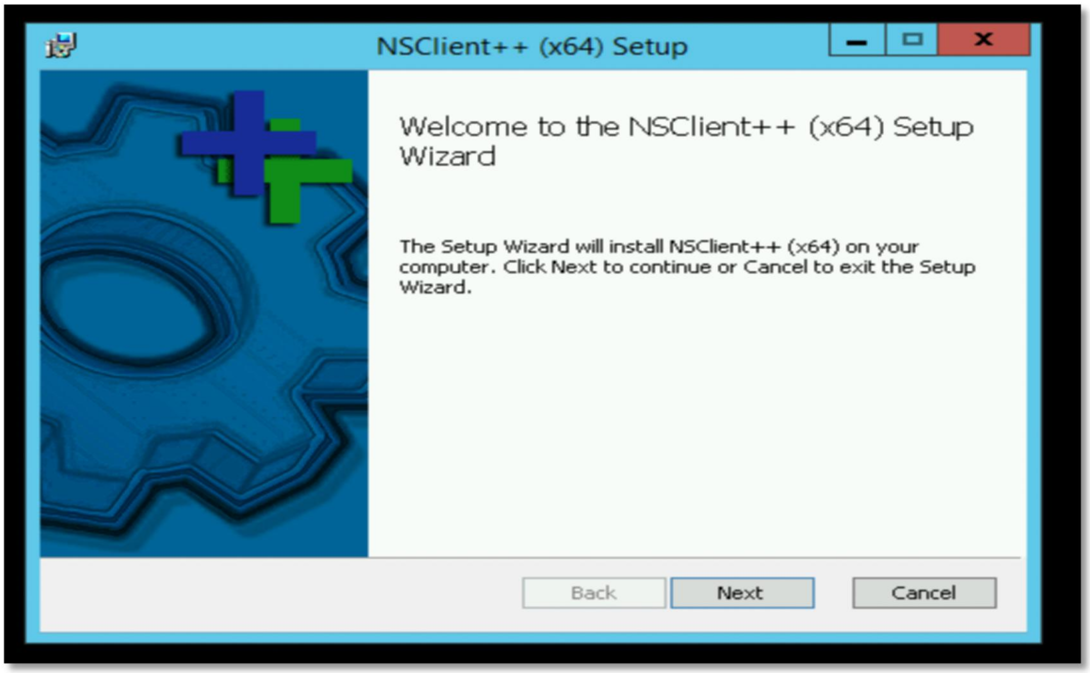


0.5.2.35

See github for details

Feel free to browse other 0.5.2 releases

Install Nsclient++ in your Windows Server.

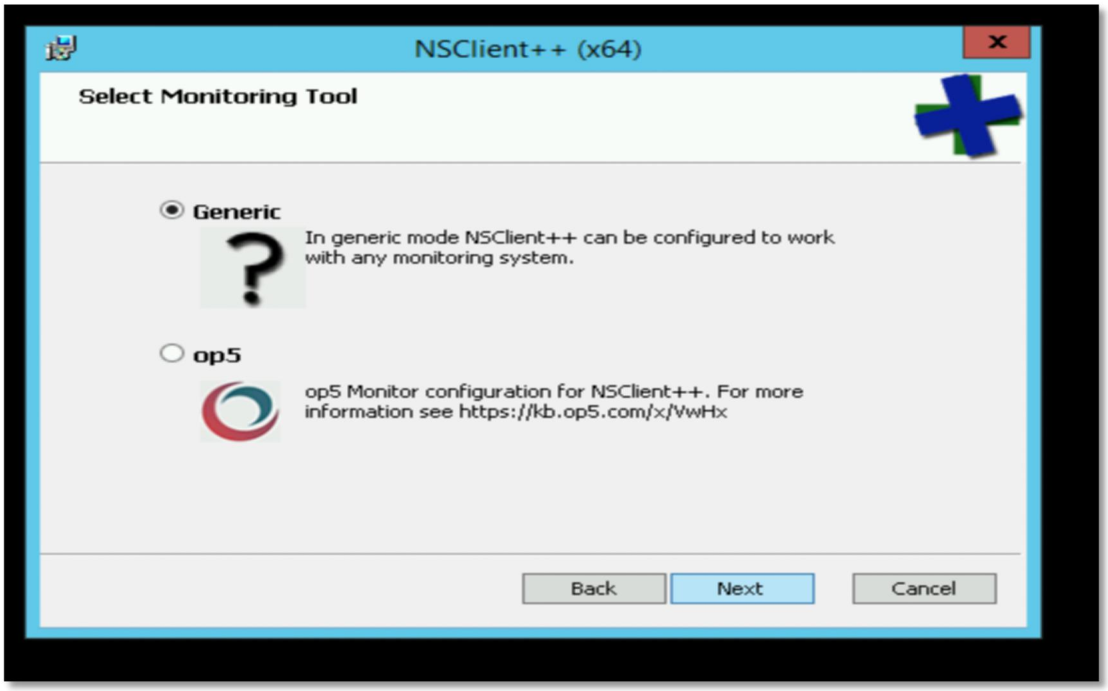


NSClient++ (x64) Setup

Welcome to the NSClient++ (x64) Setup Wizard

The Setup Wizard will install NSClient++ (x64) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Select Generic



NSClient++ (x64)

Select Monitoring Tool

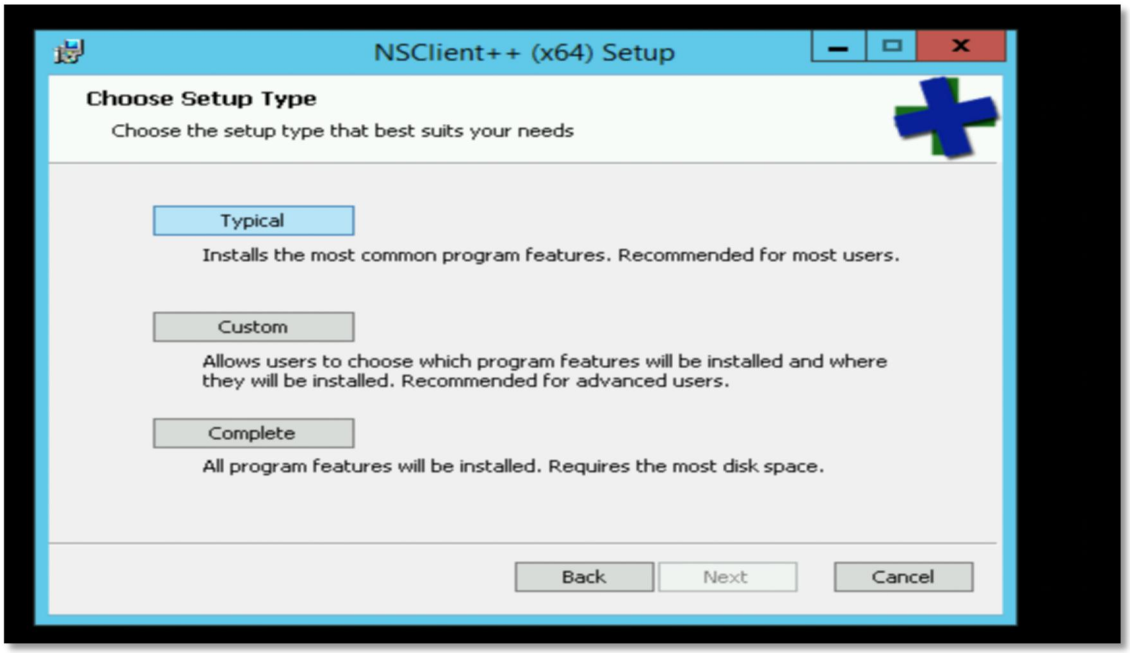
● Generic

In generic mode NSClient++ can be configured to work with any monitoring system.

○ op5

op5 Monitor configuration for NSClient++. For more information see https://kb.op5.com/x/VwHx

Select Typical



Choose Setup Type

Choose the setup type that best suits your needs

Typical

Installs the most common program features. Recommended for most users.

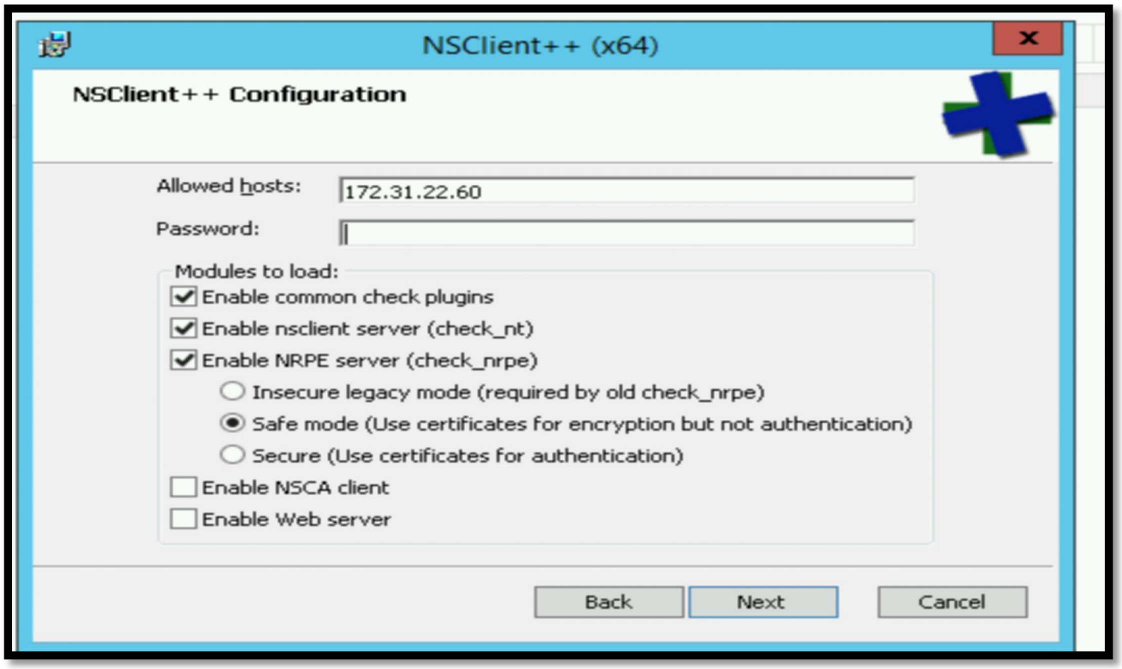
Custom

Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.

Complete

All program features will be installed. Requires the most disk space.

Enter Nagios Server IP Address in Allowed Hosts and tick mark the modules as mentioned in the below screenshot.



NSClient++ Configuration

Allowed hosts: 172.31.22.60

Password:

Modules to load:

☑ Enable common check plugins

☑ Enable nsclient server (check_nt)

☑ Enable NRPE server (check_nrpe)

○ Insecure legacy mode (required by old check_nrpe)

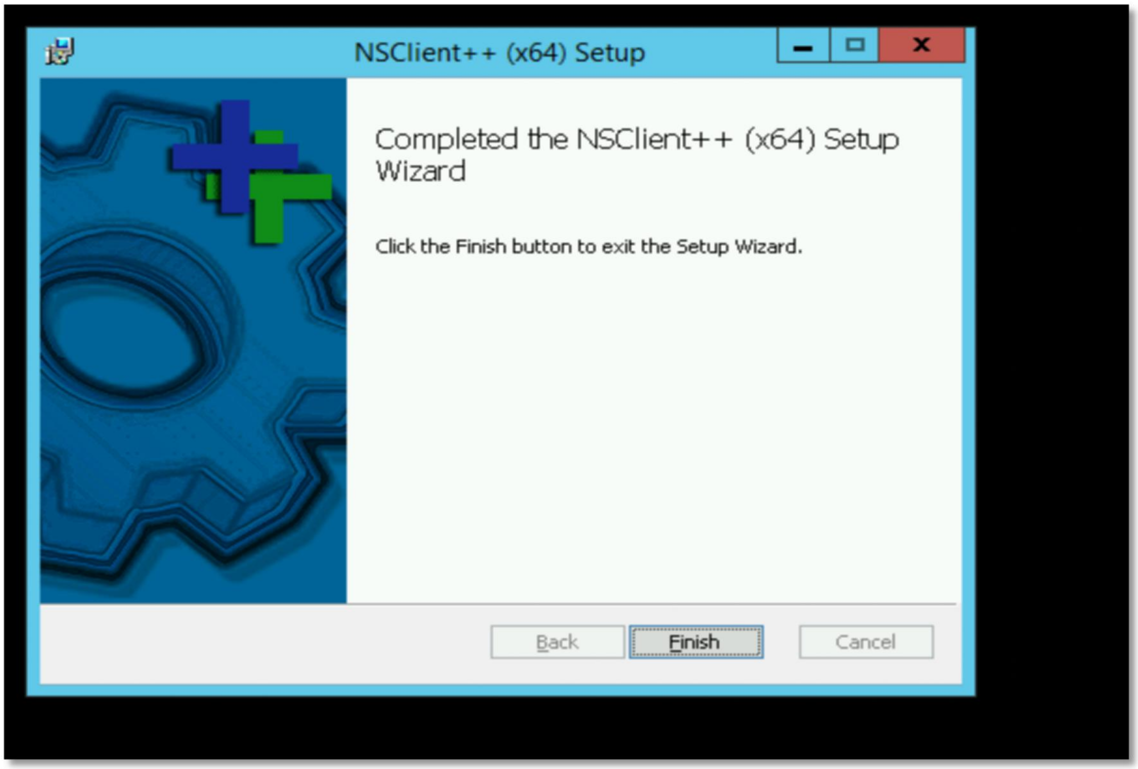
● Safe mode (Use certificates for encryption but not authentication)

○ Secure (Use certificates for authentication)

□ Enable NSCA client

□ Enable Web server

Click on Finish



NSClient++ (x64) Setup

Completed the NSClient++ (x64) Setup Wizard

Click the Finish button to exit the Setup Wizard.

nsclient.ini settings

Now open the following file as run as administrator in your Windows Server

C:\Program Files\NSClient++\nsclient.ini

CheckExternalScripts = enabled

CheckHelpers = enabled

CheckEventLog = enabled

CheckNSCP = enabled

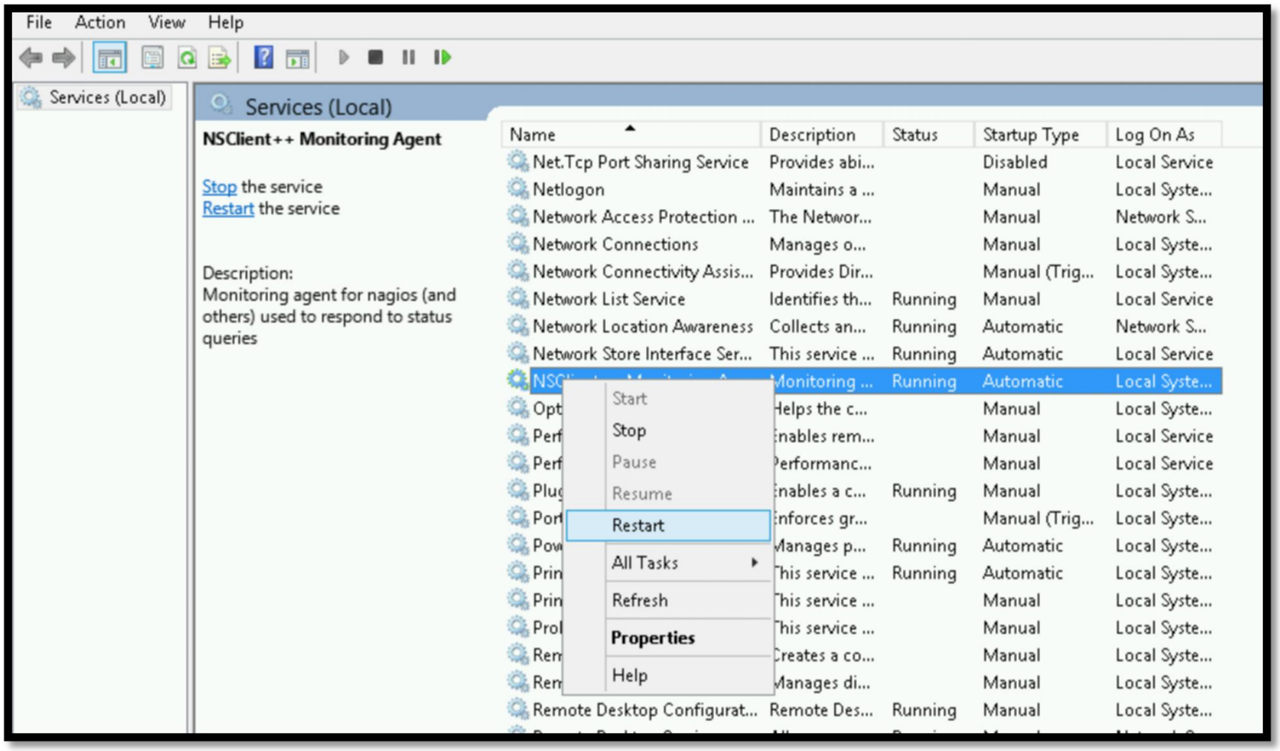
CheckDisk = enabled

CheckSystem = enabled

NSClientServer = enabled

NRPEServer = enabled

After changes restart the nsclient++ service in services.



NSClient++ Monitoring Agent

Stop the service
Restart the service

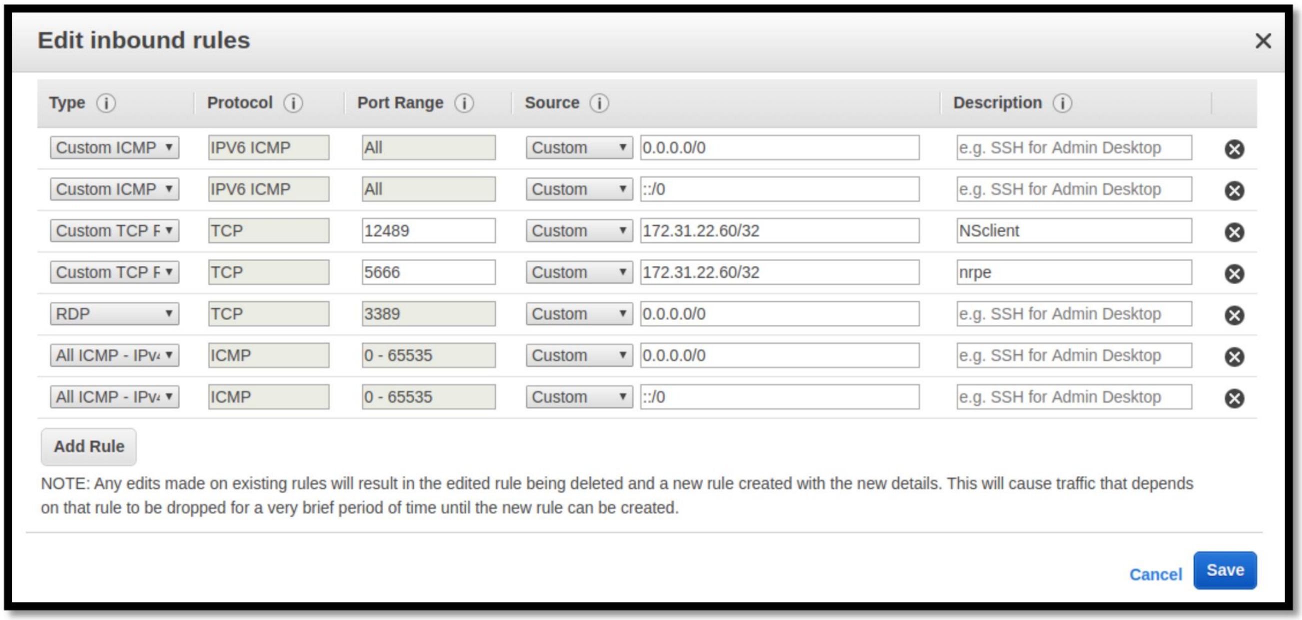
Description:
Monitoring agent for nagios (and others) used to respond to status queries

Name	Description	Status	Startup Type	Log On As
Net.Tcp Port Sharing Service	Provides abi...		Disabled	Local Service
Netlogon	Maintains a ...		Manual	Local Syste...
Network Access Protection ...	The Networ...		Manual	Network S...
Network Connections	Manages o...		Manual	Local Syste...
Network Connectivity Assis...	Provides Dir...		Manual (Trig...	Local Syste...
Network List Service	Identifies th...	Running	Manual	Local Service
Network Location Awareness	Collects an...	Running	Automatic	Network S...
Network Store Interface Ser...	This service ...	Running	Automatic	Local Service
NSC	Monitoring ...	Running	Automatic	Local Syste...
Opt	Helps the c...		Manual	Local Syste...
Perf	Enables rem...		Manual	Local Service
Perf	Performanc...		Manual	Local Service
Plug	Enables a c...	Running	Manual	Local Syste...
Port	Enforces gr...		Manual (Trig...	Local Syste...
Pow	Manages p...	Running	Automatic	Local Syste...
Prin	This service ...	Running	Automatic	Local Syste...
Prin	This service ...		Manual	Local Syste...
Prol	This service ...		Manual	Local Syste...
Rem	Creates a co...		Manual	Local Syste...
Rem	Manages di...		Manual	Local Syste...
Remote Desktop Configurat...	Remote Des...	Running	Manual	Local Syste...

We are now all done in our Windows Server.

AWS Security Group Configuration for Windows and Linux Server

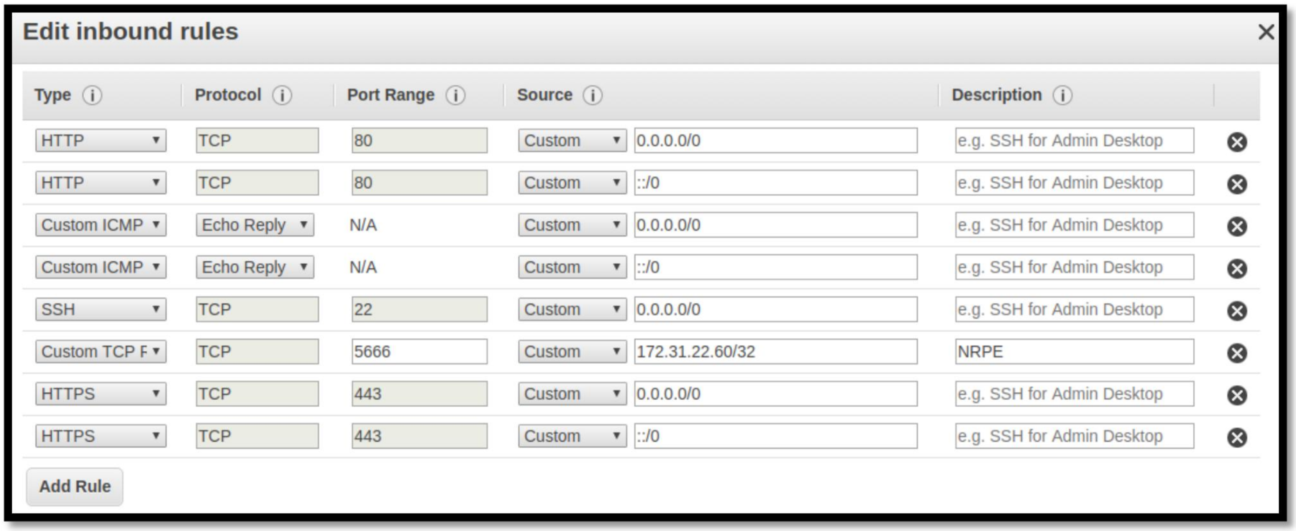
Open Security Group for Windows Server and allow port 5666 and 12489 and ICMP for Nagios Server IP.



Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ		Description ⓘ	
Custom ICMP ▾	IPV6 ICMP	All	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
Custom ICMP ▾	IPV6 ICMP	All	Custom ▾	::/0	e.g. SSH for Admin Desktop	⊗
Custom TCP F ▾	TCP	12489	Custom ▾	172.31.22.60/32	NSclient	⊗
Custom TCP F ▾	TCP	5666	Custom ▾	172.31.22.60/32	nrpe	⊗
RDP ▾	TCP	3389	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
All ICMP - IPv4 ▾	ICMP	0 - 65535	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
All ICMP - IPv4 ▾	ICMP	0 - 65535	Custom ▾	::/0	e.g. SSH for Admin Desktop	⊗

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Open Security Group for Linux Server and allow port 5666 and ICMP port for Nagios Server IP.



Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ		Description ⓘ	
HTTP ▾	TCP	80	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
HTTP ▾	TCP	80	Custom ▾	::/0	e.g. SSH for Admin Desktop	⊗
Custom ICMP ▾	Echo Reply ▾	N/A	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
Custom ICMP ▾	Echo Reply ▾	N/A	Custom ▾	::/0	e.g. SSH for Admin Desktop	⊗
SSH ▾	TCP	22	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
Custom TCP F ▾	TCP	5666	Custom ▾	172.31.22.60/32	NRPE	⊗
HTTPS ▾	TCP	443	Custom ▾	0.0.0.0/0	e.g. SSH for Admin Desktop	⊗
HTTPS ▾	TCP	443	Custom ▾	::/0	e.g. SSH for Admin Desktop	⊗

Note: If your servers are not in the AWS environment, you can allow these ports in the local firewall of both the servers.

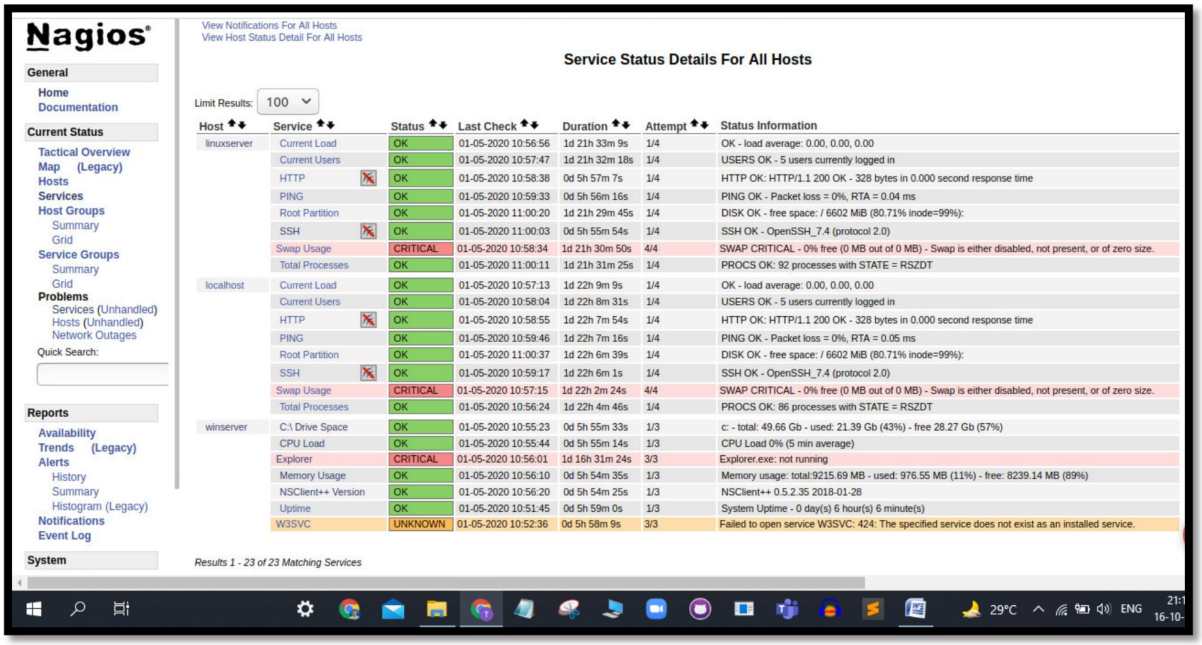
Monitor Windows and Linux Host

Now your both Linux and Windows Servers are ready to Monitor. You can monitor your servers using the following URL.

http://NagiosServerPublicIP/nagios

Default Username: nagiosadmin

Default Password : nagiosadmin



Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linuxserver	Current Load	OK	01-05-2020 10:56:56	1d 21h 33m 9s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	01-05-2020 10:57:47	1d 21h 32m 18s	1/4	USERS OK - 5 users currently logged in
	HTTP	OK	01-05-2020 10:58:38	0d 5h 57m 7s	1/4	HTTP OK: HTTP/1.1 200 OK - 328 bytes in 0.000 second response time
	PING	OK	01-05-2020 10:59:33	0d 5h 56m 16s	1/4	PING OK - Packet loss = 0%, RTA = 0.04 ms
	Root Partition	OK	01-05-2020 11:00:20	1d 21h 29m 45s	1/4	DISK OK - free space: / 6602 MiB (80.71% inode=99%):
	SSH	OK	01-05-2020 11:00:03	0d 5h 55m 54s	1/4	SSH OK - OpenSSH_7.4 (protocol 2.0)
	Swap Usage	CRITICAL	01-05-2020 10:58:34	1d 21h 30m 50s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	01-05-2020 11:00:11	1d 21h 31m 25s	1/4	PROCS OK: 92 processes with STATE = RSZDT
localhost	Current Load	OK	01-05-2020 10:57:13	1d 22h 9m 9s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	01-05-2020 10:58:04	1d 22h 8m 31s	1/4	USERS OK - 5 users currently logged in
	HTTP	OK	01-05-2020 10:58:55	1d 22h 7m 54s	1/4	HTTP OK: HTTP/1.1 200 OK - 328 bytes in 0.000 second response time
	PING	OK	01-05-2020 10:59:46	1d 22h 7m 16s	1/4	PING OK - Packet loss = 0%, RTA = 0.05 ms
	Root Partition	OK	01-05-2020 11:00:37	1d 22h 6m 39s	1/4	DISK OK - free space: / 6602 MiB (80.71% inode=99%):
	SSH	OK	01-05-2020 10:59:17	1d 22h 6m 1s	1/4	SSH OK - OpenSSH_7.4 (protocol 2.0)
	Swap Usage	CRITICAL	01-05-2020 10:57:15	1d 22h 2m 24s	4/4	SWAP CRITICAL - 0% free (0 MB out of 0 MB) - Swap is either disabled, not present, or of zero size.
	Total Processes	OK	01-05-2020 10:56:24	1d 22h 4m 46s	1/4	PROCS OK: 86 processes with STATE = RSZDT
winserver	C:\ Drive Space	OK	01-05-2020 10:55:23	0d 5h 55m 33s	1/3	c: - total: 49.66 Gb - used: 21.39 Gb (43%) - free 28.27 Gb (57%)
	CPU Load	OK	01-05-2020 10:55:44	0d 5h 55m 14s	1/3	CPU Load 0% (5 min average)
	Explorer	CRITICAL	01-05-2020 10:56:01	1d 16h 31m 24s	3/3	Explorer.exe: not running
	Memory Usage	OK	01-05-2020 10:56:10	0d 5h 54m 35s	1/3	Memory usage: total:9215.69 MB - used: 976.55 MB (11%) - free: 8239.14 MB (89%)
	NSClient++ Version	OK	01-05-2020 10:56:20	0d 5h 54m 25s	1/3	NSClient++ 0.5.2.35 2018-01-28
	Uptime	OK	01-05-2020 10:51:45	0d 5h 59m 0s	1/3	System Uptime - 0 day(s) 6 hour(s) 6 minute(s)
	W3SVC	UNKNOWN	01-05-2020 10:52:36	0d 5h 58m 9s	3/3	Failed to open service W3SVC: 424: The specified service does not exist as an installed service.

Results 1 - 23 of 23 Matching Services

Conclusion:

Hence, We successfully performed Port, Service monitoring, Windows/Linux server monitoring using Nagios.