**NAGIOS CORE INSTALLATION ON UBUNTU 20.04 FROM THE SOURCE**

<https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source96.html#Ubuntu>

apt update && apt upgrade

sudo apt install -y build-essential apache2 php openssl perl make php-gd libgd-dev libapache2 mod-php libperl-dev libssl-dev daemon wget apache2-utils unzip

sudo useradd nagios

sudo groupadd nagcmd

sudo usermod -a -G nagcmd nagios

sudo usermod -a -G nagcmd www-data

cd /tmp

wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.5.tar.gz

tar -zxvf /tmp/nagios-4.4.5.tar.gz

cd /tmp/nagios-4.4.5/

sudo ./configure --with-nagios-group=nagios --with-command-group=nagcmd --with-httpd\_conf=/etc/apache2/sites-enabled/

sudo make all

sudo make install

sudo make install-init

sudo make install-config

sudo make install-commandmode

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

# set password: cteladmin

sudo make install-webconf

sudo a2enmod cgi

sudo systemctl restart apache2

cd /tmp

wget https://nagios-plugins.org/download/nagios-plugins-2.3.3.tar.gz

tar -zxvf /tmp/nagios-plugins-2.3.3.tar.gz

cd /tmp/nagios-plugins-2.3.3/

sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios

sudo make

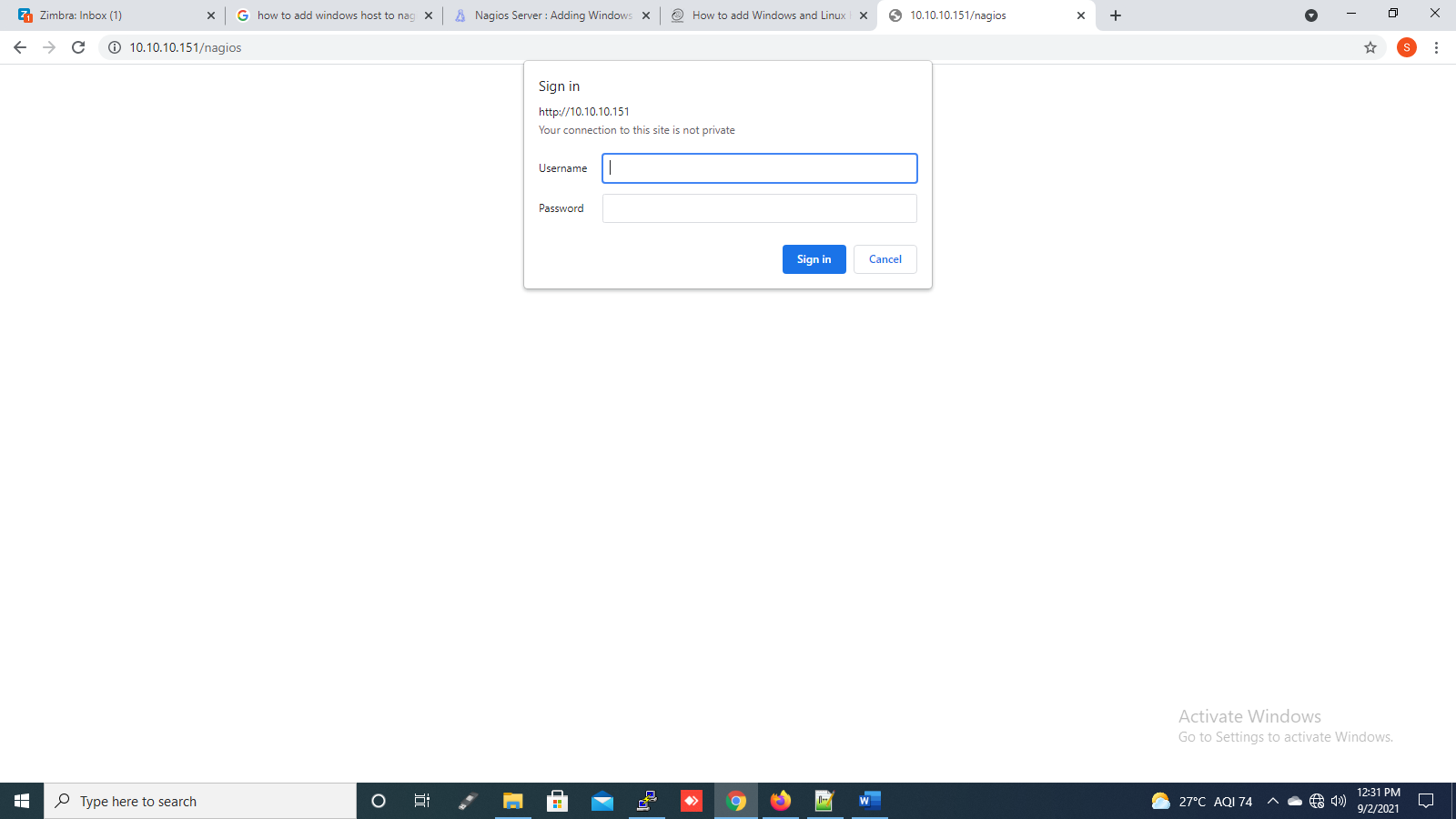
sudo make install

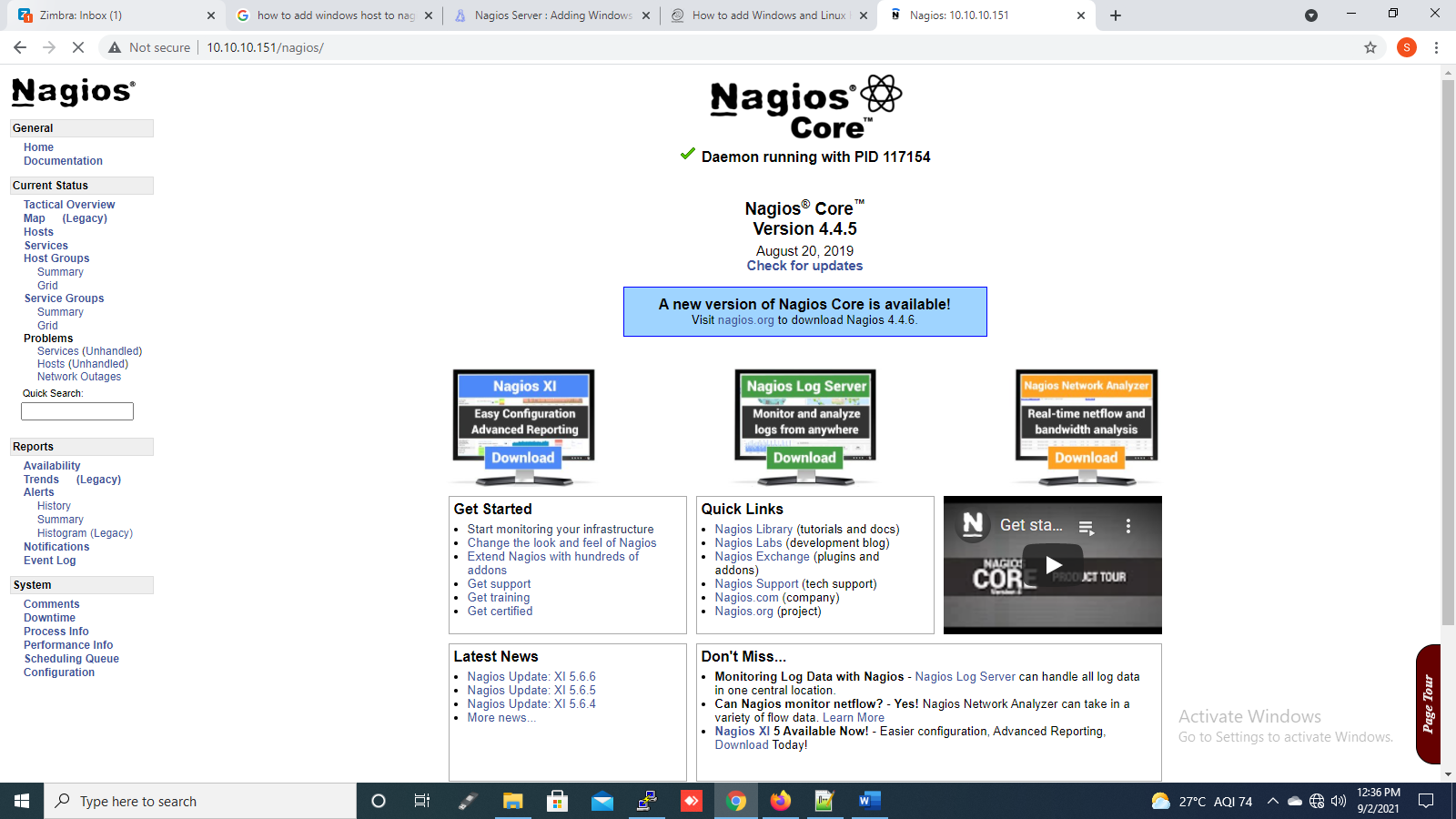
sudo /usr/local/nagios/bin/nagios -v

vim /usr/local/nagios/etc/nagios.cfg

sudo systemctl enable nagios

sudo systemctl start nagios





**NRPE PLUGINS**

**Prerequisites**

//Make sure that you have the following packages installed.

sudo apt-get update

sudo apt-get install -y autoconf automake gcc libc6 libmcrypt-dev make libssl-dev wget openssl

//Downloading the Source

cd /tmp

wget --no-check-certificate -O nrpe.tar.gz <https://github.com/NagiosEnterprises/nrpe/archive/nrpe-4.0.3.tar.gz>

tar xzf nrpe.tar.gz

//Compile

Note that if you want to pass arguments through NRPE you must specify this in the configuration option as indicated below. If you prefer to you can omit the --enable-command-args flag. Removing this flag will require that all arguments be explicitly set in the nrpe.cfg file on each server monitored.

cd /tmp/nrpe-nrpe-4.0.3/

sudo ./configure --enable-command-args --with-ssl-lib=/usr/lib/x86\_64-linux-gnu/

sudo make all

//Create User And Group

//This creates the nagios user and group.

sudo make install-groups-users

//Install Binaries

//This step installs the binary files, the NRPE daemon and the check\_nrpe plugin.

If you only wanted to install the daemon, run the command make install-daemon instead of the command below. However it useful having the check\_nrpe plugin installed for testing purposes.

If you only wanted to install the check\_nrpe plugin, refer to the section at the bottom of this KB article as there a lot of steps that can be skipped. Installing only the plugin is usually done on your Nagios server and workers.

sudo make install

//Install Configuration Files

//This installs the config files.

sudo make install-config

//Update Services File

//The /etc/services file is used by applications to translate human readable service names into port numbers when connecting to a machine across a network.

sudo sh -c "echo >> /etc/services"

sudo sh -c "sudo echo '# Nagios services' >> /etc/services"

sudo sh -c "sudo echo 'nrpe 5666/tcp' >> /etc/services"

//Install Service / Daemon

sudo make install-init

sudo systemctl enable nrpe.service

Information on starting and stopping services will be explained further on. Configure Firewall

Port 5666 is used by NRPE and needs to be opened on the local firewall.

sudo mkdir -p /etc/ufw/applications.d

sudo sh -c "echo '[NRPE]' > /etc/ufw/applications.d/nagios"

sudo sh -c "echo 'title=Nagios Remote Plugin Executor' >> /etc/ufw/applications.d/nagios"

sudo sh -c "echo 'description=Allows remote execution of Nagios plugins' >> /etc/ufw/applications.d/nagios"

sudo sh -c "echo 'ports=5666/tcp' >> /etc/ufw/applications.d/nagios"

sudo ufw allow NRPE

sudo ufw reload

//Update Configuration File

The file nrpe.cfg is where the following settings will be defined. It is located:

/usr/local/nagios/etc/nrpe.cfg

allowed\_hosts= to allowed\_hosts=127.0.0.1,10.10.10.151

At this point NRPE will only listen to requests from itself (127.0.0.1). If you wanted your nagios server to be able to connect, add it's IP address after a comma (in this example it's 10.25.5.2):

dont\_blame\_nrpe=0 to dont\_blame\_nrpe=1

This option determines whether or not the NRPE daemon will allow clients to specify arguments to commands that are executed. We are going to allow this, as it enables more advanced NPRE configurations.

//The following commands make the configuration changes described above.

sudo sh -c "sed -i '/^allowed\_hosts=/s/$/,10.10.10.152/' /usr/local/nagios/etc/nrpe.cfg"

sudo sh -c "sed -i 's/^dont\_blame\_nrpe=.\*/dont\_blame\_nrpe=1/g' /usr/local/nagios/etc/nrpe.cfg"

//Start Service / Daemon

sudo systemctl start nrpe.service

//Test NRPE, Now check that NRPE is listening and responding to requests.

/usr/local/nagios/libexec/check\_nrpe -H 127.0.0.1

Output: NRPE v4.0.3

If you get the NRPE version number (as shown above), NRPE is installed and configured correctly.

You can also test from your Nagios host by executing the same command above, but instead of 127.0.0.1 you will need to replace that with the IP Address / DNS name of the machine with NRPE running.

//Service / Daemon Commands

sudo systemctl start nrpe.service

sudo systemctl stop nrpe.service

sudo systemctl restart nrpe.service

sudo systemctl status nrpe.service

**configurations:**

vim /usr/local/nagios/etc/objects/localhost.cfg

Add the hostnames and hostname groups for services also

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 localhost

alias localhost

address 10.10.10.151

}

define host {

use linux-server ; Name of host template to use

host\_name Ashish

alias Ashish

address 10.10.10.153

}

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 saiteja

alias saiteja

address 10.10.10.152

}

# Define an optional hostgroup for Linux machines

define hostgroup {

hostgroup\_name linux-servers ; The name of the hostgroup

alias Linux Servers ; Long name of the group

members localhost,saiteja,Ashish ; Comma separated list of hosts that belong to this group

}

#############################################################################

# SERVICE DEFINITIONS

##############################################################################

# Define a service to "ping" the local machine

define service {

use local-service ; Name of service template to use

host\_name localhost,saiteja,Ashish

service\_description PING

check\_command check\_ping!100.0,20%!500.0,60%

}

# Define a service to check the disk space of the root partition

# on the local machine. Warning if < 20% free, critical if

# < 10% free space on partition.

define service {

use local-service ; Name of service template to use

host\_name localhost,saiteja,Ashish

service\_description Root Partition

}

# Define a service to check the number of currently logged in

# users on the local machine. Warning if > 20 users, critical

# if > 50 users.

define service {

use local-service ; Name of service template to use

host\_name localhost,saiteja,Ashish

service\_description Current Users

check\_command check\_local\_users!20!50

}

define service {

use local-service ; Name of service template to use

host\_name localhost,saiteja,Ashish

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 localhost,saiteja,Ashish

service\_description Current Load

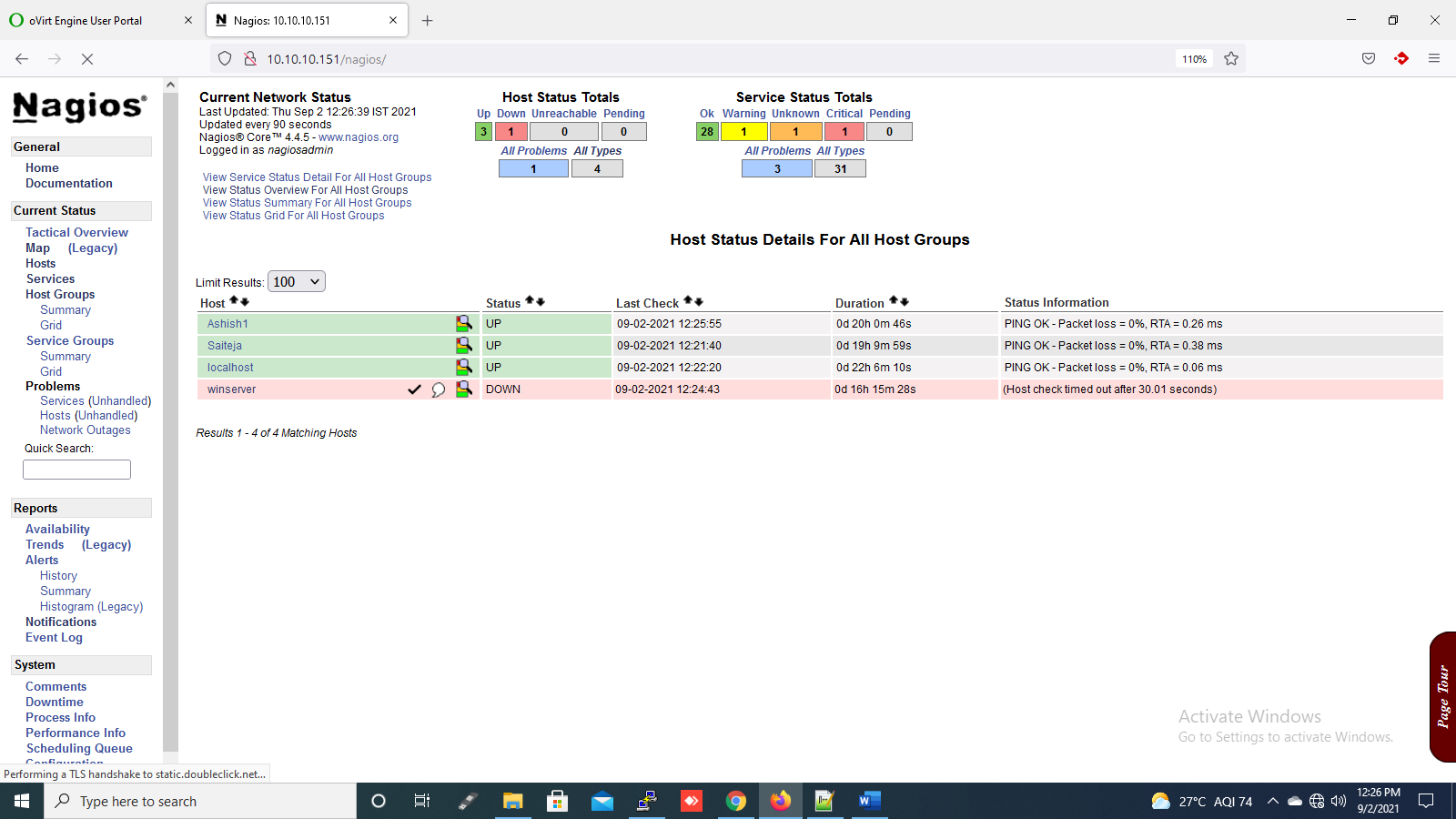
check\_command check\_local\_load!5.0,4.0,3.0!10.0,6.0,4.0

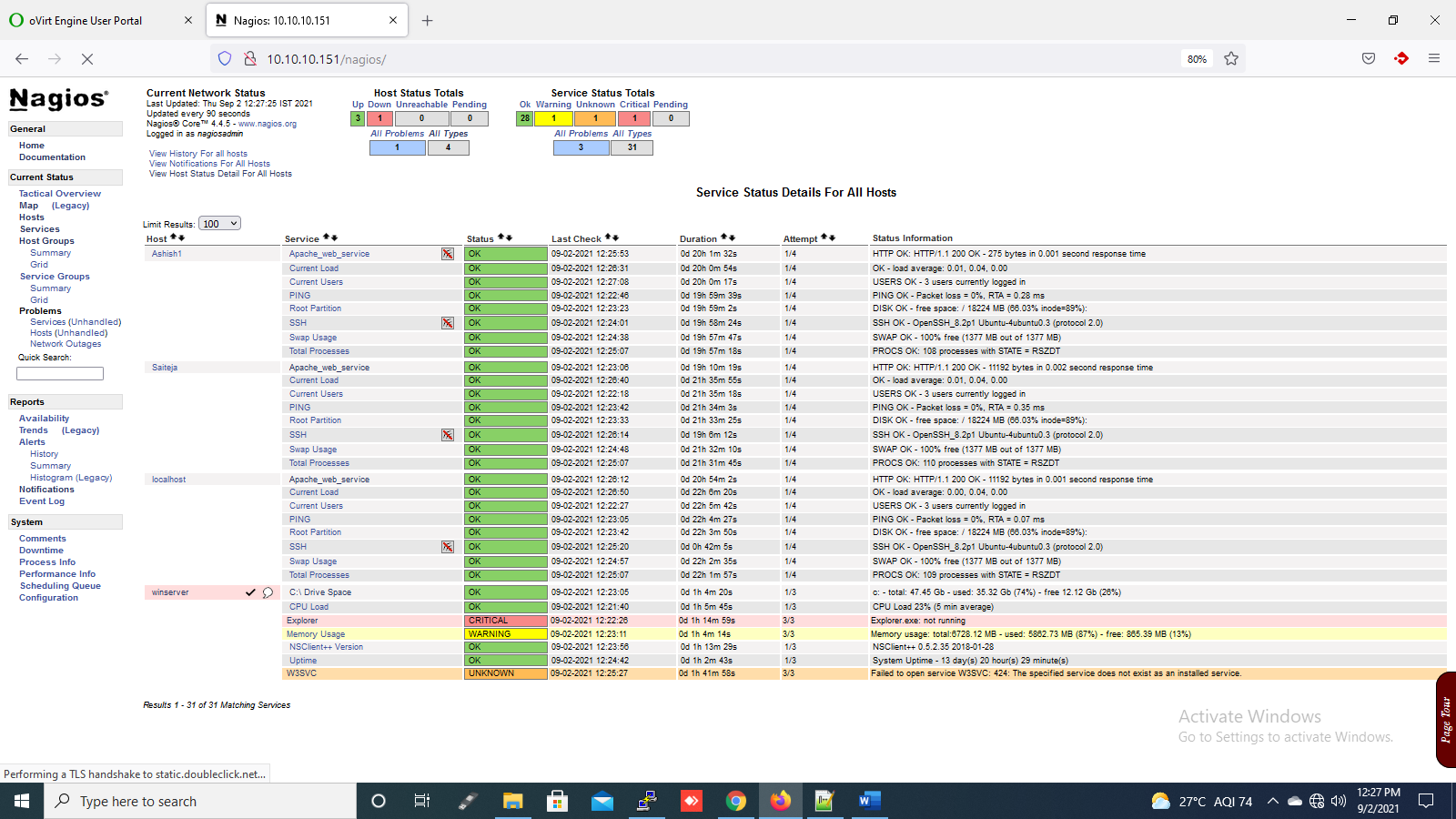
}

....

....

....





**FOR ALERTS:**

Initially configure SMTP server to send mails from your server

vim /usr/local/nagios/etc/objects/contacts.cfg

##############################################################################

# CONTACTS

##############################################################################

# Just one contact defined by default - the Nagios admin (that's you)

# This contact definition inherits a lot of default values from the

# 'generic-contact' template which is defined elsewhere.

define contact {

contact\_name nagiosadmin ; Short name of user

use generic-contact ; Inherit default values from generic-contact template (defined above)

alias Nagios Admin ; Full name of user

email saikiran.k@ctel.in ; <<\*\*\*\*\* CHANGE THIS TO YOUR EMAIL ADDRESS \*\*\*\*\*\*>>

}

**ADDING HOST IN NAGIOS SERVER:**

sudo nano /usr/local/nagios/etc/servers/<hostfilename.cfg>

# Define a host for the local machine

define host {

use linux-server ; Name of host template to use

host\_name Ashish1

alias Ashish1

address 10.10.10.153

}

# Define a service to "ping" the local machine

define service {

use local-service ; Name of service template to use

host\_name Ashish1

service\_description PING

check\_command check\_ping!100.0,20%!500.0,60%

}

# Define a service to check the disk space of the root partition

# on the local machine. Warning if < 20% free, critical if

# < 10% free space on partition.

define service {

use local-service ; Name of service template to use

host\_name Ashish1

service\_description Root Partition

check\_command check\_local\_disk!20%!10%!/

}

# Define a service to check the number of currently logged in

# users on the local machine. Warning if > 20 users, critical

# if > 50 users.

define service {

use local-service ; Name of service template to use

host\_name Ashish1

service\_description Current Users

check\_command check\_local\_users!20!50

}

Save and quit ( :wq )

next add the path cfgfile=/usr/local/nagios/etc/servers/<hostfilename.cfg> in /usr/local/nagios/etc/nagios.cfg

systemctl restart nagios.service

systemctl restart nrpe.service

**NAGIOS WINDOWS HOST ADDING TO NAGIOS SERVER**

Install the NSclient++ plugin in windows machine (It acts as a agent in windows)

**IN WINDOWS HOST:**

Download the NSClient++ from,

http://nsclient.org/download/

Install the **NSClient++,**

Select **Typical** >> **Add the allowed hosts** (server IP's) >> **enable** all the services >> next and **install.**

Go to the NSClient file,

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

Enable the services,

# If you want to fill this file with all available options run the following command:

# nscp settings --generate --add-defaults --load-all

# If you want to activate a module and bring in all its options use:

# nscp settings --activate-module <MODULE NAME> --add-defaults

# For details run: nscp settings --help

; in flight - TODO

[/settings/default]

; Undocumented key

allowed hosts = 10.10.10.151,10.10.10.152,10.10.10.153

; in flight - TODO

[/settings/NRPE/server]

; Undocumented key

ssl options = no-sslv2,no-sslv3

; Undocumented key

verify mode = peer-cert

; Undocumented key

insecure = false

; in flight - TODO

[/modules]

; Undocumented key

CheckExternalScripts = enabled

; Undocumented key

CheckHelpers = enabled

; Undocumented key

CheckNSCP = enabled

; Undocumented key

CheckDisk = enabled

; Undocumented key

WEBServer = enabled

; Undocumented key

CheckSystem = enabled

; Undocumented key

NSClientServer = enabled

; Undocumented key

CheckEventLog = enabled

; Undocumented key

NSCAClient = enabled

; Undocumented key

NRPEServer = enabled

Task Manager >> services >> open services >> NSclient >> stop/ restart

**IN NAGIOS SERVER LINUX MACHINE:**

sudo vim /usr/local/nagios/etc/objects/windows.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 10.10.14.156 ; IP address of the host

}

##################################################################

# SERVICE DEFINITIONS ##################################################################

# Create a service for monitoring the version of NSCLient++ that is installed

# Change the host\_name to match the name of the host you defined above

define service {

use generic-service

host\_name winserver

service\_description NSClient++ Version

check\_command check\_nt!CLIENTVERSION

}

# Create a service for monitoring the uptime of the server

# Change the host\_name to match the name of the host you defined above

define service {

use generic-define service {

use generic-service

host\_name winserver

service\_description Uptime

check\_command check\_nt!UPTIME

}

sudo vim /usr/local/nagios/etc/nagioscfg

Add line:

cfgfile=/usr/local/nagios/etc/objects/windows.cfg

systemctl restart nagios.service

systemctl restart nrpe.service

**NAGIOS INSTALLATION ON UBUNTU 20.04**

<https://tecadmin.net/how-to-install-nagios-server-on-ubuntu-20-04/>

sudo apt update

sudo apt install apache2

sudo a2enmod authz\_groupfile auth\_digest

sudo apt update

sudo apt install nagios4 nagios-nrpe-plugin nagios-plugins-contrib

sudo htdigest -c /etc/nagios4/htdigest.users Nagios4 nagiosadmin

Give password:

sudo nano /etc/apache2/conf-enabled/nagios4-cgi.conf

Do this things:

Comment Require IP line

Files start and closing tag only

Comment “Required all granted”

Uncomment “Require valid-user”

sudo nano /etc/nagios4/cgi.cfg

Change use\_authentication to 1:

use\_authentication=1

sudo systemctl restart apache2

sudo systemctl restart nagios4

http://server\_domain\_or\_ip/nagios4

**NAGIOS NRPE PLUGINS INSTALLATION ON UBUNTU 20.04**

sudo apt update

sudo apt upgrade

sudo apt install nagios-nrpe-server nagios-plugins

sudo systemctl status nagios-nrpe-server

sudo nano /etc/nagios/nrpe.cfg

Edit the file and give the allowed IP's (Nagiosserver IP)

allowed\_hosts=127.0.0.1, 10.10.10.151,10.10.10.152

sudo systemctl restart nagios-nrpe-server

check\_nrpe -H 192.168.1.11

sudo firewall-cmd --permanent --zone=public --add-port=5666/tcp

sudo firewall-cmd --reload