HOW TO INSTALL NAGIOS, CACTI AND SPEEDTEST MINI ON UBUNTU SERVER 16.04

On the Ubuntu Server install LAMP during the server installation.

CREATE SSL CERTIFICATES

root@snet[~]# cd /etc/ssl/private

root@snet[~]# openssl genrsa -aes128 -out server.key 2048

OUTPUT

Generating RSA private key, 2048 bit long modulus

.......+++..+++e is 65537 (0x10001)

Enter pass phrase for server.key: # set passphrase

Verifying - Enter pass phrase for server.key: # confirm

root@snet[~]# openssl rsa -in server.key -out server.key

OUTPUT

Enter pass phrase for server.key: # passphrase writing RSA key

root@snet[~]# openssl req -new -days 3650 -key server.key -out server.csr

OUTPUT

Country Name (2 letter code) [AU]: GH   # country

State or Province Name (full name) [Some-State]: ACCRA   # state

Locality Name (eg, city) []: ACCRA   # city

Organization Name (eg, company) [Internet Widgits Pty Ltd]: STEP NETWORK # company

Organizational Unit Name (eg, section) []: CLOUD SOLUTIONS # department

Common Name (e.g. server FQDN or YOUR name) []: sdrive.snetgh.net #
server's FQDN

Email Address []: info@snetgh.net   # email address

Please enter the following 'extra' attributes to be sent with your certificate
request

A challenge password []:

An optional company name []

root@snet[~]# openssl x509 -in server.csr -out server.crt -req -signkey server.key -
days 3650

root@snet[~]# chmod 400 server.*

root@snet[~]# nano /etc/apache2/sites-available/default-ssl.conf

 # Line 3: change admin email

ServerAdmin info@snetgh.net

 # Line 32, 33: change to the one created above

SSLCertificateFile /etc/ssl/private/server.crt

SSLCertificateKeyFile /etc/ssl/private/server.key

root@snet[~]# a2ensite default-ssl

root@snet[~]# a2enmod ssl

root@snet[~]# systemctl restart apache2

OPTIONAL CONFIGURATION REDIRECT HTTP TO HTTPS

root@snet[~]# nano /etc/apache2/sites-available/000-default.conf

*__Add a redirect directive under <VirtualHost *:80>__*

Redirect "/" "https://172.168.1.80/"

root@snet[~]# sudo a2enmod ssl

root@snet[~]# sudo a2enmod headers

root@snet[~]# a2ensite default-ssl

root@snet[~]# sudo service apache2 restart

INSTALLATION COMMANDS

root@snet[~]# sudo nano /etc/apt/sources.list

Add the following line at the bottom of the file:

root@snet[~]# deb http://download.webmin.com/download/repository sarge contrib

root@snet[~]# wget http://www.webmin.com/jcameron-key.asc

root@snet[~]# sudo apt-key add jcameron-key.asc

root@snet[~]# sudo apt-get update

root@snet[~]# sudo apt-get install webmin

INSTALLATION OF UBUNTU SERVER PACKAGES

root@snet[~]# apt-get install build-essential apache2 php7.0 openssl perl make php7.0-gd libgd2-xpm-dev libapache2-mod-php7.0 libperl-dev libssl-dev daemon wget apache2-utils unzip

root@snet[~]# sudo apt-get install lamp-server^

root@snet[~]# sudo apt-get install php7.0-mcrypt

root@snet[~]# sudo add-apt-repository ppa:ondrej/php

root@snet[~]# sudo apt-get update

root@snet[~]# sudo apt-get install php7.0-gd

root@snet[~]# sudo apt-get install php7.0-intl

root@snet[~]# sudo apt-get update

root@snet[~]# sudo apt-get install php-xml

root@snet[~]# sudo add-apt-repository -y ppa:ondrej/php && sudo apt update && apt install -y php7.0-mbstring php7.0-zip php7.0-xml

root@snet[~]# sudo apt-get install php7.0-zip

root@snet[~]# sudo apt-get install php7.0-curl

root@snet[~]# sudo apt-get install unzip

root@snet[~]# apt-get install php-apcu php-apcu-bc

INSTALLATION OF PHPMYADMIN

phpMyAdmin is one of the most popular and widely used web-based database management tools. It a free and open source PHP application that allows the users to manage single or multiple SQL database servers locally or on a remote server using a web browser (GUI).

INSTALLATION COMMANDS

root@snet[~]# apt-get -y install phpmyadmin php-mbstring php-gettext

root@snet[~]# sudo systemctl restart apache2

SECURE YOUR PHPMYADMIN

root@snet[~]# sudo nano /etc/apache2/conf-available/phpmyadmin.conf

To Configure Apache's .htaccess files

We need to add an AllowOverride All **Then find the line where there is**

<Directory /usr/share/phpmyadmin>

Options FollowSymLinks

DirectoryIndex index.php

AllowOverride All

root@snet[~]# sudo systemctl restart apache2

root@snet[~]# sudo nano /usr/share/phpmyadmin/.htaccess

Add the following information

AuthType Basic

AuthName "Restricted Files"

AuthUserFile /etc/phpmyadmin/.htpasswd

Require valid-user

root@snet[~]# sudo apt-get install apache2-utils

ADDING A USER TO THE PHPMYADMIN AUTHENTICATION

root@snet[~]# sudo htpasswd -c /etc/phpmyadmin/.htpasswd snet

Adding Another User

root@snet[~]# sudo htpasswd /etc/phpmyadmin/.htpasswd stepehen

HOW TO CONFIGURE APACHE .HTACCESS FOR MAXIMUM FILE UPLOAD

root@snet[~]# sudo nano /etc/apache2/apache2.conf

To Configure Apache's .htaccess files

Change **'AllowOverride None**' to '**AllowOveride All**'

Then find the line where there is

```
<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```
Replace "None" with "All"

```
AllowOverride All
```

Next, press CTRL+O, press enter, and then press CTRL+X to exit.

Note: Reboot the Ubuntu server

CREATING OF DATABASE FOR OWNCLOUD SERVER

root@snet[~]# mysql -u root -p

root@snet[~]# CREATE DATABASE owncloud;

root@snet[~]# GRANT ALL ON owncloud.* to 'owncloud'@'localhost' IDENTIFIED BY 'stephen';

root@snet[~]# FLUSH PRIVILEGES;

root@snet[~]# exit

OR

Navigate to phpmyadmin by http://server's name or IP/phpmyadmin and create database name called "**owncloud**" and click on create that's all. Very easier than using the console or above.

Using PuTTY to remote the server and issue the following commands to install nagios and cacti:

<u>INSTALLATION OF NAGIOS</u>

root@snet[~]# apt-get update

root@snet[~]# apt-get install build-essential apache2 php7.0 openssl perl make php7.0-gd libgd2-xpm-dev libapache2-mod-php7.0 libperl-dev libssl-dev daemon wget apache2-utils unzip

root@snet[~]# useradd nagios

root@snet[~]# groupadd nagcmd

root@snet[~]# usermod -a -G nagcmd nagios

root@snet[~]# usermod -a -G nagcmd www-data

root@snet[~]# cd /tmp

root@snet[~]# wget https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.4.1.tar.gz

root@snet[~]# tar -zxvf /tmp/nagios-4.4.1.tar.gz

root@snet[~]# cd /tmp/nagios-4.4.1/

```
root@snet[~]# ./configure --with-nagios-group=nagios --with-command-
group=nagcmd --with-httpd_conf=/etc/apache2/sites-enabled/


root@snet[~]# make all


root@snet[~]# make install


root@snet[~]# make install-init


root@snet[~]# make install-config


root@snet[~]# make install-commandmode


root@snet[~]# make install-webconf


root@snet[~]# nano /usr/local/nagios/etc/objects/contacts.cfg
```
Change the email address field to receive the notification.

```
root@snet[~]# htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin


root@snet[~]# a2enmod cgi


root@snet[~]# service apache2 restart


root@snet[~]# cd /tmp
```

root@snet[~]# wget http://www.nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz

root@snet[~]# tar -zxvf /tmp/nagios-plugins-2.2.1.tar.gz

root@snet[~]# cd /tmp/nagios-plugins-2.2.1/

root@snet[~]# ./configure --with-nagios-user=nagios --with-nagios-group=nagios

root@snet[~]# make

root@snet[~]# make install

root@snet[~]# nano /etc/systemd/system/nagios.service

Add below content into above file.

[Unit]

Description=Nagios

BindTo=network.target

 [Install]

WantedBy=multi-user.target

 [Service]

User=nagios

Group=nagios

Type=simple

ExecStart=/usr/local/nagios/bin/nagios /usr/local/nagios/etc/nagios.cfg

root@snet[~]# systemctl enable /etc/systemd/system/nagios.service

root@snet[~]# systemctl start nagios

INSTALLATION OF CACTI

root@snet[~]# sudo apt-get install cacti cacti-spine

ADDING NEW HOSTS TO NAGIOS SERVER

root@snet[~]# sudo nano /usr/local/nagios/etc/objects/new_host_name.cfg

And adding the following lines to it

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   google.com

    alias   google.com

    address   www.google.com

    }

```
###############################################################################
###############################################################################
#
# SERVICE DEFINITIONS
#
###############################################################################
###############################################################################


# Define a service to "ping" the local machine


define service{
        use                     generic-service         ; Name of service template to use
        host_name               google.com
        service_description         PING
        check_command               check_ping!100.0,20%!500.0,60%
        }


# Define a service to check HTTP on the local machine.
# Disable notifications for this service by default, as not all users may have HTTP enabled.


define service{
```

```
    use                    generic-service        ; Name of service template to use
    host_name              google.com
    service_description        HTTP
    check_command            check_http
    notifications_enabled      0
    }
```

root@snet[~]# sudo nano /usr/local/nagios/etc/nagios.cfg

Add the folowwing line to the file

cfg_file=/usr/local/nagios/etc/objects/newhost.cfg

root@snet[~]# sudo systemctl start nagios

root@snet[~]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

HOW TO ADD WINDOWS CLIENT TO NAGIOS

root@snet[~]# cd /usr/local/nagios/etc/

root@snet[~]# touch windows-hosts.cfg

root@snet[~]# touch windows-services.cfg

root@snet[~]# nano /usr/local/nagios/etc/nagios.cfg

Copy the following files to the bottom

cfg_file=/usr/local/nagios/etc/windows-hosts.cfg
cfg_file=/usr/local/nagios/etc/windows-services.cfg

root@snet[~]# nano /usr/local/nagios/etc/objects/windows.cfg

"Now u can edit your host and services"

root@snet[~]# nano /usr/local/nagios/etc/nagios.cfg

Uncomment line 38 and then save

root@snet[~]# service nagios restart

root@snet[~]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

HOW TO DEFINE NEW GROUP IN NAGIOS SERVER

root@snet[~]# nano /usr/local/nagios/etc/objects/templates.cfg

#Define a template for cloud-server that can be reuse

define host{

name	cloud-server; The name of this host template	
use template	generic-host ; Inherit default values from the generic-host	
check_period clock	24x7 ; By default, Windows servers are monitored round the	
check_interval	5; Actively check the server every 5 minutes	
retry_interval	1 ; Schedule host check retries at 1 minute intervals	
max_check_attempts	10 ; Check each server 10 times (max)	
check_command are "alive"	check-host-alive ; Default command to check if servers	
notification_period	24x7; Send notification out at any time - day or night	
notification_interval	30 ; Resend notifications every 30 minutes	
notification_options	d,r ; Only send notifications for specific host states	
contact_groups	admins ; Notifications get sent to the admins by default	
hostgroups be a member of	cloud-servers ; Host groups that Windows servers should	
register	0 ; DONT REGISTER THIS - ITS JUST A TEMPLATE	

}

sudo nano /usr/local/nagios/etc/objects/cloud.cfg

Define Hosts

define host{

use   cloud-server; Inherit default values from a template

host_name   cserver1 ; The name we're giving to this host

alias   Cloud Server 1 ; A longer name associated with the host

address   192.168.1.1 ; IP address of the host

contacts   nagiosadmin

}

Define host groups

define hostgroup{

hostgroup_name   cloud-servers

alias   Cloud Servers

members   cserver1
}

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

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

#

SERVICE DEFINITIONS

#

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

```
################################################################
############

# Define a service to "ping" the local machine
define service{
        use                     generic-service      ; Name of service template to use
        host_name               cloud
        service_description     PING
        check_command            check_ping!100.0,20%!500.0,60%
        }


# Define a service to check HTTP on the local machine.
# Disable notifications for this service by default, as not all users may have HTTP
enabled.
define service{
        use                     generic-service      ; Name of service template to use
        host_name               cloud
        service_description     HTTP
        check_command            check_http
        notifications_enabled     0
        }


# Define a service to check SSH on the local machine.# Disable notifications for
this service by default, as not all users may have SSH enabled.
define service{
```

```
        use                     generic-service        ; Name of service template to use

        host_name               cloud

        service_description     SSH

        check_command            check_ssh

        notifications_enabled     0

        }
```

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

```
define service{

        use                     generic-service        ; Name of service template to use

        host_name               cloud

        service_description      Current Load

        check_command            check_local_load!5.0,4.0,3.0!10.0,6.0,4.0

        }
```

Define a service to check the number of currently running procs

on the local machine. Warning if > 250 processes, critical if

> 400 processes.

```
define service{

        use                     generic-service        ; Name of service template to use

        host_name               cloud

        service_description      Total Processes

        check_command            check_local_procs!250!400!RSZDT

        }
```

root@snet[~]# sudo nano /usr/local/nagios/etc/nagios.cfg

Add the folowwing line to the file

cfg_file=/usr/local/nagios/etc/objects/cloud.cfg

root@snet[~]# sudo systemctl start nagios

root@snet[~]# sudo /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

To check errors of your configuration

We have to install NSClient++ on the windows Host

http://www.nsclient.org/download/

<u>WEBSITE TO BE VISITED FOR MORE INFORMATION</u>

http://www.the-tech-tutorial.com/adding-hosts-to-nagios/

Cloud Storage host definition template - This is NOT a real host, just a template!

```
define host{

    name                    cloud-servers   ; The name of this host template

    use                     generic-host    ; This template inherits other values from
the generic-host template

    check_period            24x7            ; By default, Linux hosts are checked
round the clock

    check_interval          5               ; Actively check the host every 5 minutes

    retry_interval          1               ; Schedule host check retries at 1 minute
intervals

    max_check_attempts      10              ; Check each Linux host 10 times
(max)

    check_command           check-host-alive ; Default command to check
Linux hosts

    notification_period     workhours       ; Linux admins hate to be woken
up, so we only notify during the day

                            ; Note that the notification_period variable is
being overridden from

                            ; the value that is inherited from the generic-
host template!

    notification_interval   120             ; Resend notifications every 2 hours

    notification_options    d,u,r           ; Only send notifications for specific
host states

    contact_groups          admins          ; Notifications get sent to the admins
by default

    register                0               ; DONT REGISTER THIS DEFINITION -
ITS NOT A REAL HOST, JUST A TEMPLATE!

    }
```