Deep Chavan

T11-15

LAB ASSIGNMENT 9

AIM: Installation of nagios on ubuntu system.

LAB OUTCOME:
LO1, LO5 Mapped.

THEORY:
Nagios is an open-source monitoring and alerting system that is widely used to monitor the availability and health of IT infrastructure components, including servers, network devices, applications, and services. It helps organisations maintain the reliability and stability of their systems by providing real time visibility into the performance and status of various components in their environment.

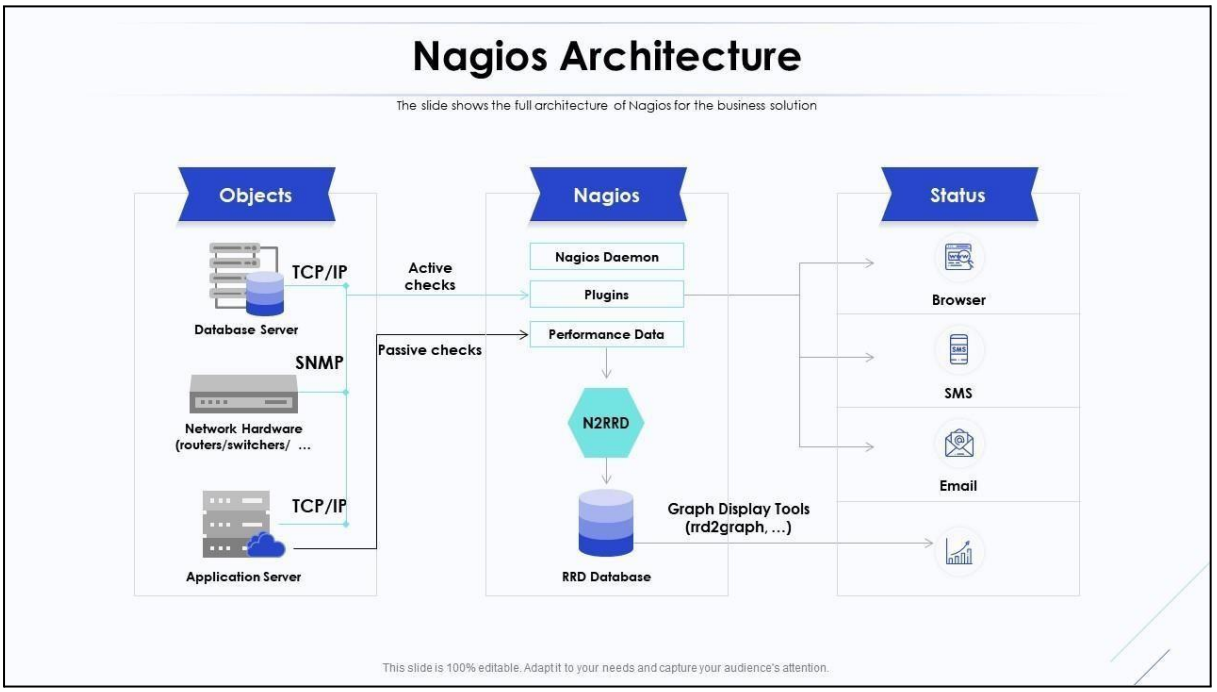
Key features and components of Nagios includes:

1. **Host and Service Monitoring:** Nagios allows users to define hosts (e.g., servers, routers) and services (e.g., web services, email servers) to be monitored. It periodically checks these hosts and services to ensure they are functioning correctly.
2. **Alerting and Notification**: When Nagios detects a problem or a service outage, it can send notifications to designated administrators or teams through various methods, including email, SMS, and custom scripts. This enables timely responses to issues.
3. **Flexible Configuration:** Nagios is highly configurable and allows users to define custom checks, thresholds, and notification rules. This flexibility makes it suitable for a wide range of monitoring scenarios.
4. **Web Interface:** Nagios provides a web-based dashboard that offers a real-time view of the monitored infrastructure's status. Administrators can access this dashboard to see which services are up or down and view historical performance data.
5. **Plugin Architecture:** Nagios uses a plugin system that allows users to extend its monitoring capabilities. Many plugins are available for monitoring specific applications, devices, or protocols, and users can develop custom plugins as needed.
6. **Performance Graphs**: Nagios can collect performance data and display it in graphs and charts.

This helps in analysing historical trends and identifying potential issues before they become critical.

7. **Event Logging**: Nagios keeps a detailed log of monitoring events and notifications. This log can be useful for troubleshooting and auditing.
8. **Scheduled Downtime**: Administrators can schedule downtime for planned maintenance or upgrades to prevent unnecessary alerts during maintenance windows.
9. **Community and Support**: Nagios has an active user community, which provides resources, documentation, and support for users. There are also commercial versions and third-party tools built around Nagios for additional features and support.

Nagios is highly versatile and can be used in various IT environments, from small businesses to large enterprises. It plays a crucial role in ensuring the availability and performance of critical infrastructure components, helping organisations proactively address issues and minimise downtime. Additionally, Nagios can be integrated into larger IT management and monitoring solutions to provide comprehensive visibility into an organisation's technology stack.



WORKING:

Nagios works based on a simple yet effective principle: monitoring and alerting. It continuously checks the status and performance of various hosts (e.g., servers, network devices) and services (e.g., applications, websites) by running predefined checks, known as plugins.

1. **Configuration:** The Nagios administrator defines what needs to be monitored and how in a configuration file. This includes specifying hosts, services, notification settings, and alert thresholds. Users can configure checks to run at specific intervals, such as every minute, and set warning and critical thresholds for each service (e.g., response time should be under 200ms).

2. **Checks and Plugins**: Nagios periodically runs these checks based on the defined intervals and uses plugins to perform the checks. A plugin is a small script or executable that carries out a specific monitoring task, such as pinging a server, checking the HTTP response code of a website, or monitoring disk space usage.

3. **Status Data**: After running the checks, Nagios collects status data, including the results of the checks, timestamps, and performance metrics (if applicable). This data is stored internally.

4. **Alerting and Notifications**: If a check indicates a problem (e.g., a service is down, a server is unresponsive, a threshold is exceeded), Nagios triggers an alert. Alerts can be notifications sent to administrators or teams through various means, such as email, SMS, or custom scripts. These alerts inform relevant personnel about the issue, enabling them to take action.

5. **Dashboard and Reports**: Nagios provides a web-based dashboard where administrators can view the real-time status of monitored hosts and services. They can see which services are up, which are down, and view historical performance data in the form of graphs and charts. This dashboard is a central hub for monitoring and managing the infrastructure.

6. **Event Logging**: Nagios keeps a detailed log of all monitoring events, including check results, alerts, and notifications. This log serves as an audit trail and can be useful for troubleshooting and historical analysis.

7. **Scheduled Downtime**: Nagios allows administrators to schedule downtime for hosts and services during planned maintenance windows. This prevents Nagios from generating unnecessary alerts during maintenance activities.

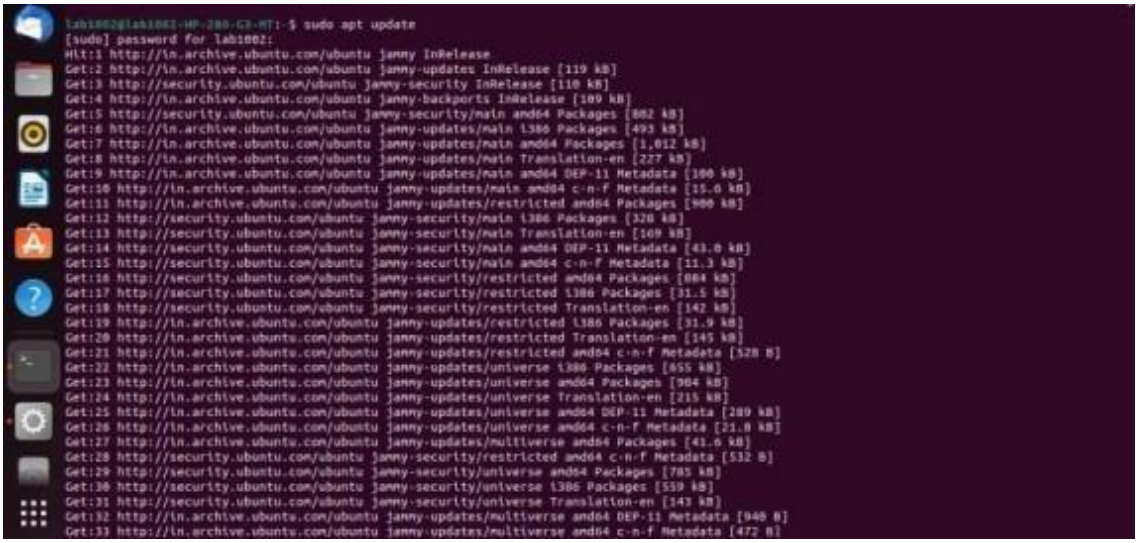
8. **Recovery Notifications:** When a previously failed service or host returns to a healthy state, Nagios can send recovery notifications to inform administrators that the issue has been resolved.

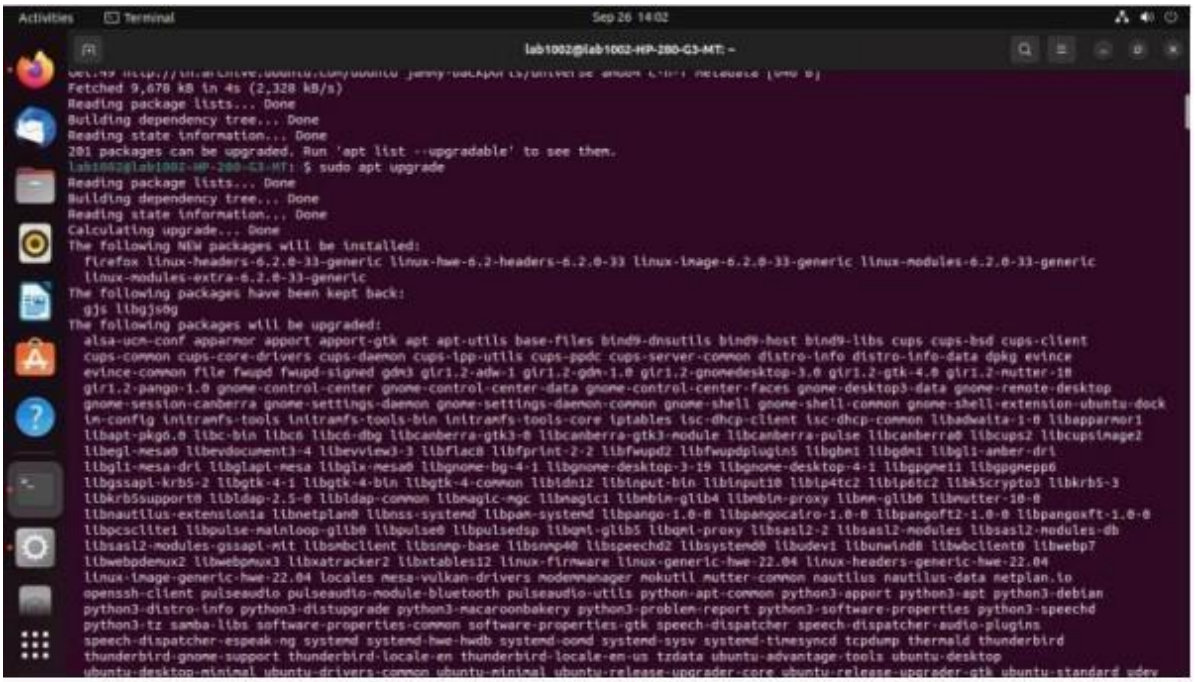
9. **Escalation and Acknowledgements**: Nagios supports advanced features like escalation, where alerts can be escalated to higher-level teams if not acknowledged or resolved within a certain timeframe. Administrators can also acknowledge alerts, indicating that they are aware of the issue and are working on it.

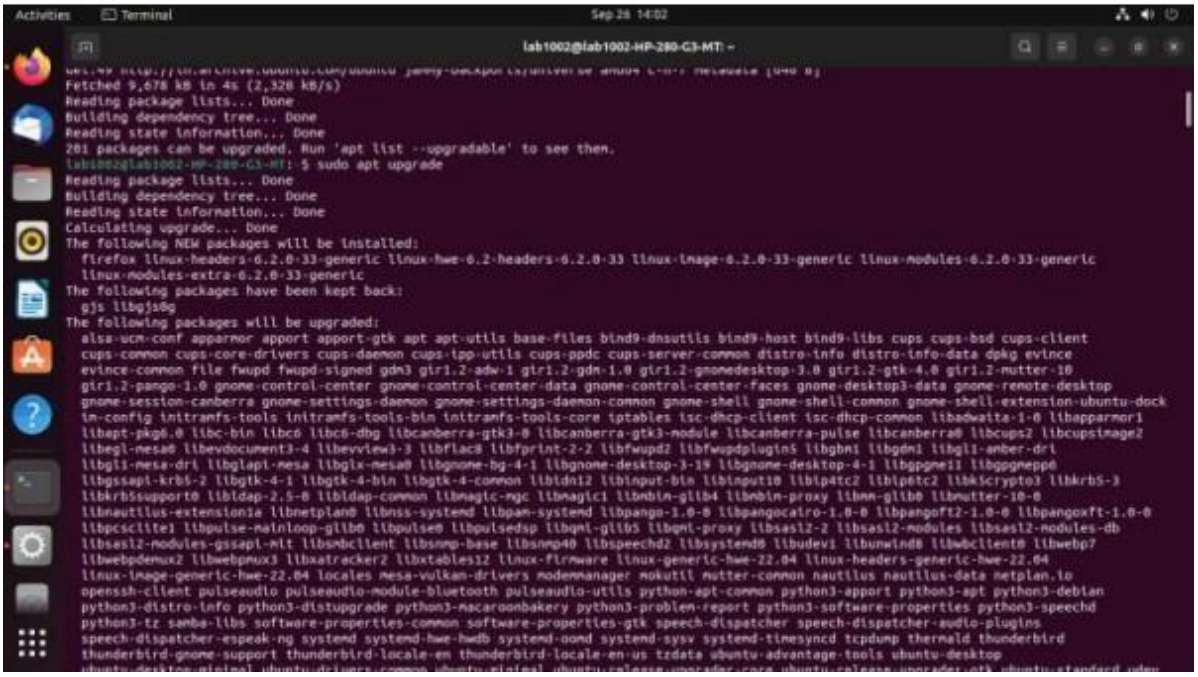
10. **Performance Data:** Nagios can collect and display performance data, which helps in identifying trends, bottlenecks, and potential issues before they become critical. This data can be used for capacity planning and optimization.

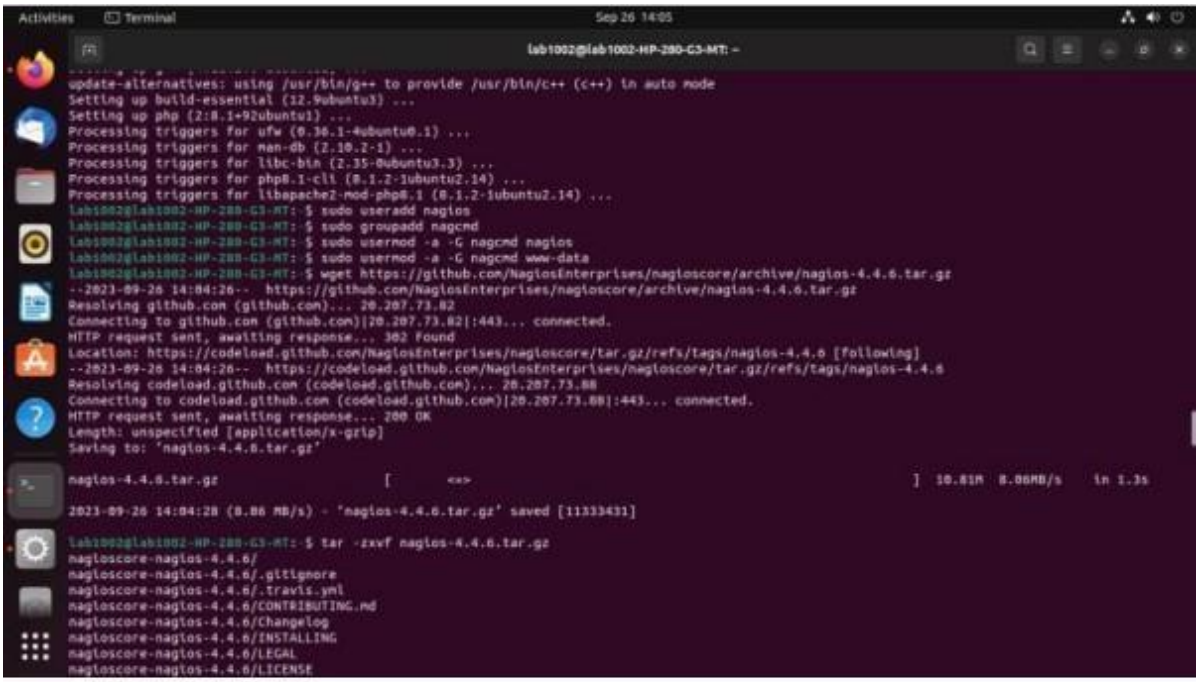
Nagios operates continuously, providing real-time monitoring and alerting for IT infrastructure components. It helps organisations maintain the availability and performance of their systems and applications while enabling prompt responses to issues, ultimately reducing downtime and ensuring a more stable IT environment.

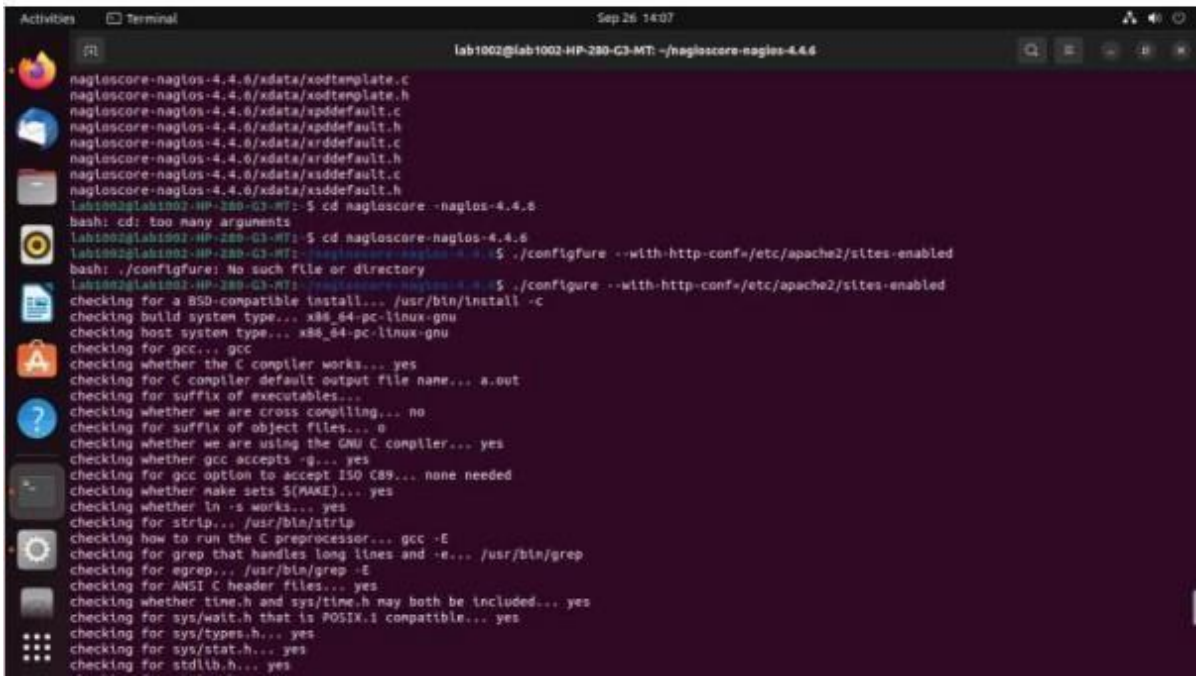
INSTALLATION STEPS:

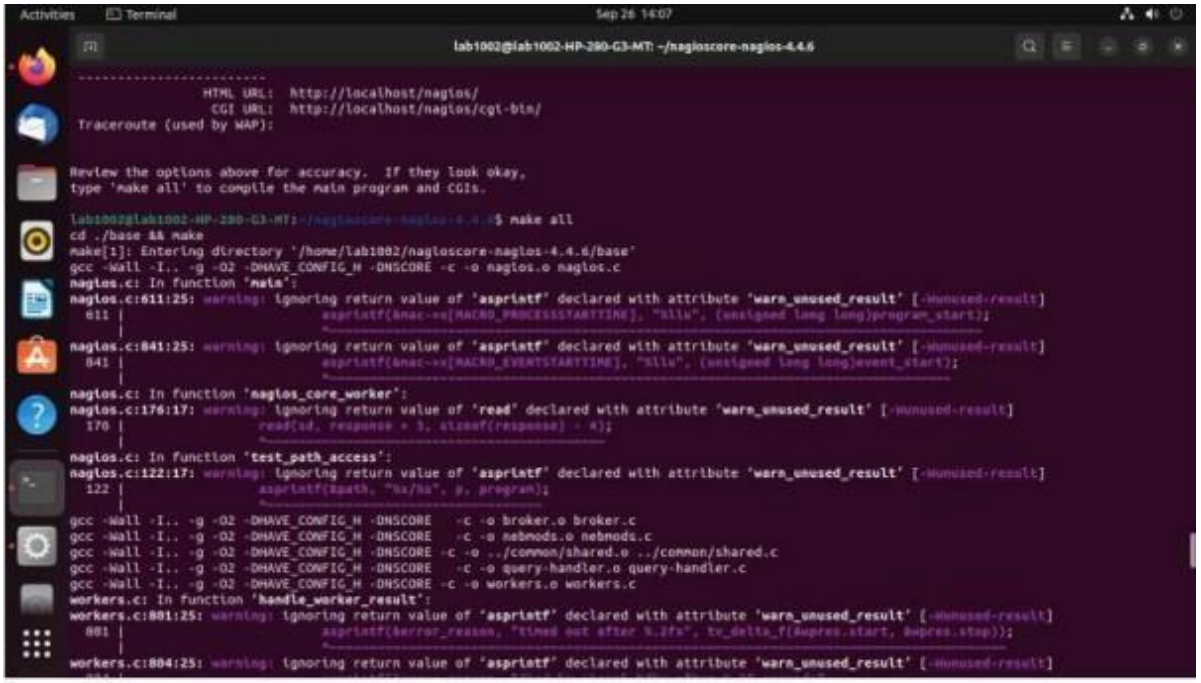


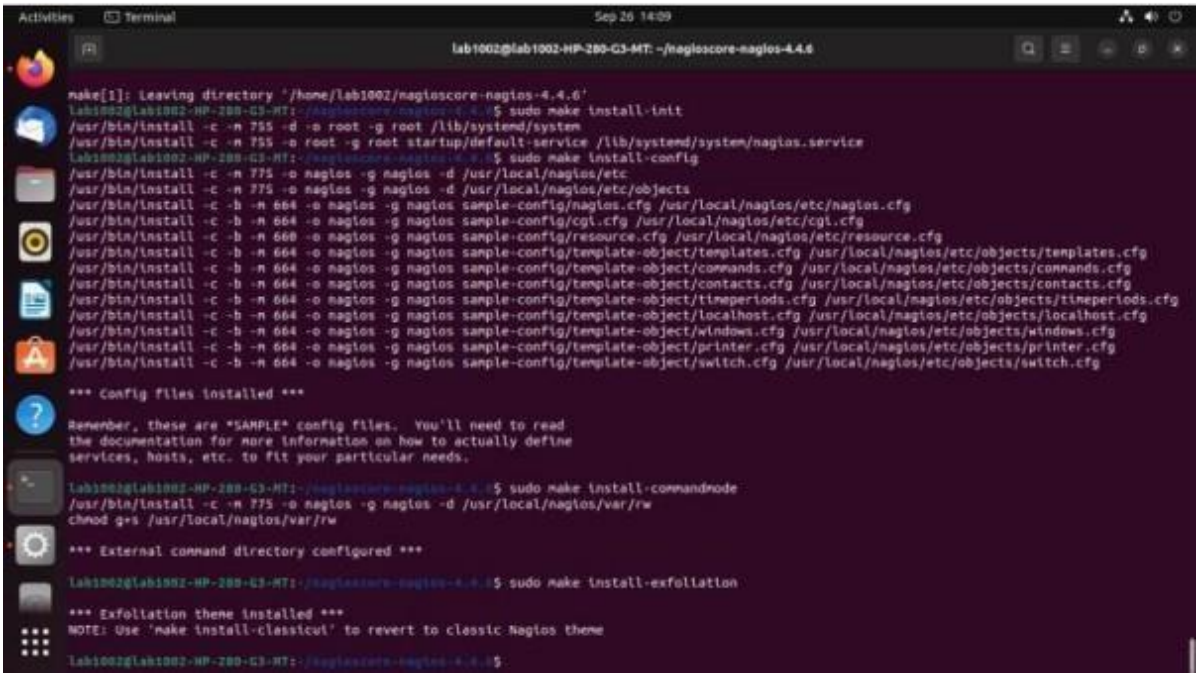

Activities
Terminal
Sep 26 14:02
lab1002@lab1002-HP-280-G3-MT: ~
Fetched 9,670 kB in 4s (2,328 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
201 packages can be upgraded. Run 'apt list --upgradable' to see them.
lab1002@lab1002-HP-280-G3-MT: $ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
firefox linux-headers-6.2.0-33-generic linux-hwe-6.2-headers-6.2.0-33 linux-image-6.2.0-33-generic linux-modules-6.2.0-33-generic
linux-modules-extra-6.2.0-33-generic
The following packages have been kept back:
gjs libgjs0g
The following packages will be upgraded:
alsa-ucm-conf apparmor apport apport-gtk apt apt-utils base-files bind9-dnsutils bind9-host bind9-libs cups cups-bsd cups-client
cups-common cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc cups-server-common distro-info distro-info-data dpkg evince
evince-common file fwupd fwupd-signed gdm3 gir1.2-adw-1 gir1.2-gdm-1.0 gir1.2-gnomedesktop-3.0 gir1.2-gtk-4.0 gir1.2-nutter-10
gir1.2-pango-1.0 gnome-control-center gnome-control-center-data gnome-control-center-faces gnome-desktop3-data gnome-remote-desktop
gnome-session-canberra gnome-settings-daemon gnome-settings-daemon-common gnome-shell gnome-shell-common gnome-shell-extension-ubuntu-dock
im-config initramfs-tools initramfs-tools-bin initramfs-tools-core iptables isc-dhcp-client isc-dhcp-common libadwaita-1-0 libapparmor1
libapt-pkg6.0 libc-bin libc6 libc6-dbg libcanberra-gtk3-0 libcanberra-gtk3-module libcanberra-pulse libcanberra0 libcups2 libcupsimage2
libegl-mesa0 libevdocument3-4 libevview3-3 libflac8 libfprint-2-2 libfwupd2 libfwupdplugin5 libgbm1 libgdm1 libgl1-amber-dri
libgl1-mesa-dri libglapi-mesa libglx-mesa0 libgnome-bg-4-1 libgnome-desktop-3-19 libgnome-desktop-4-1 libgpgme11 libgpgmepp6
libgssapi-krb5-2 libgtk-4-1 libgtk-4-bin libgtk-4-common libidn12 libinput-bin libinput10 liblp4tc2 libip6tc2 libk5crypto3 libkrb5-3
libkrb5support0 libldap-2.5-0 libldap-common libmagic-mgc libmagic1 libnbin-glib4 libnbin-proxy libnm-glib0 libnutter-10-0
libnautilus-extension1a libnetplan0 libnss-systemd libpam-systemd libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpangoxft-1.0-0
libpcsclite1 libpulse-mainloop-glib0 libpulse0 libpulsedsp libqmi-glib5 libqmi-proxy libsasl2-2 libsasl2-modules libsasl2-modules-db
libsasl2-modules-gssapi-mit libsmbclient libsnmp-base libsnmp40 libspeechd2 libsystemd0 libudev1 libunwind8 libwbclient0 libwebp7
libwebpdemux2 libwebpmux3 libxatracker2 libxtables12 linux-firmware linux-generic-hwe-22.04 linux-headers-generic-hwe-22.04
linux-image-generic-hwe-22.04 locales mesa-vulkan-drivers modemmanager mokutil mutter-common nautilus nautilus-data netplan.io
openssh-client pulseaudio pulseaudio-module-bluetooth pulseaudio-utils python-apt-common python3-apport python3-apt python3-debian
python3-distro-info python3-distupgrade python3-macaroonbakery python3-problem-report python3-software-properties python3-speechd
python3-tz samba-libs software-properties-common software-properties-gtk speech-dispatcher speech-dispatcher-audio-plugins
speech-dispatcher-espeak-ng systemd systemd-hwe-hwdb systemd-oomd systemd-sysv systemd-timesyncd tcpdump thermald thunderbird
thunderbird-gnome-support thunderbird-locale-en thunderbird-locale-en-us tzdata ubuntu-advantage-tools ubuntu-desktop
ubuntu-desktop-minimal ubuntu-drivers-common ubuntu-minimal ubuntu-release-upgrader-core ubuntu-release-upgrader-gtk ubuntu-standard udev

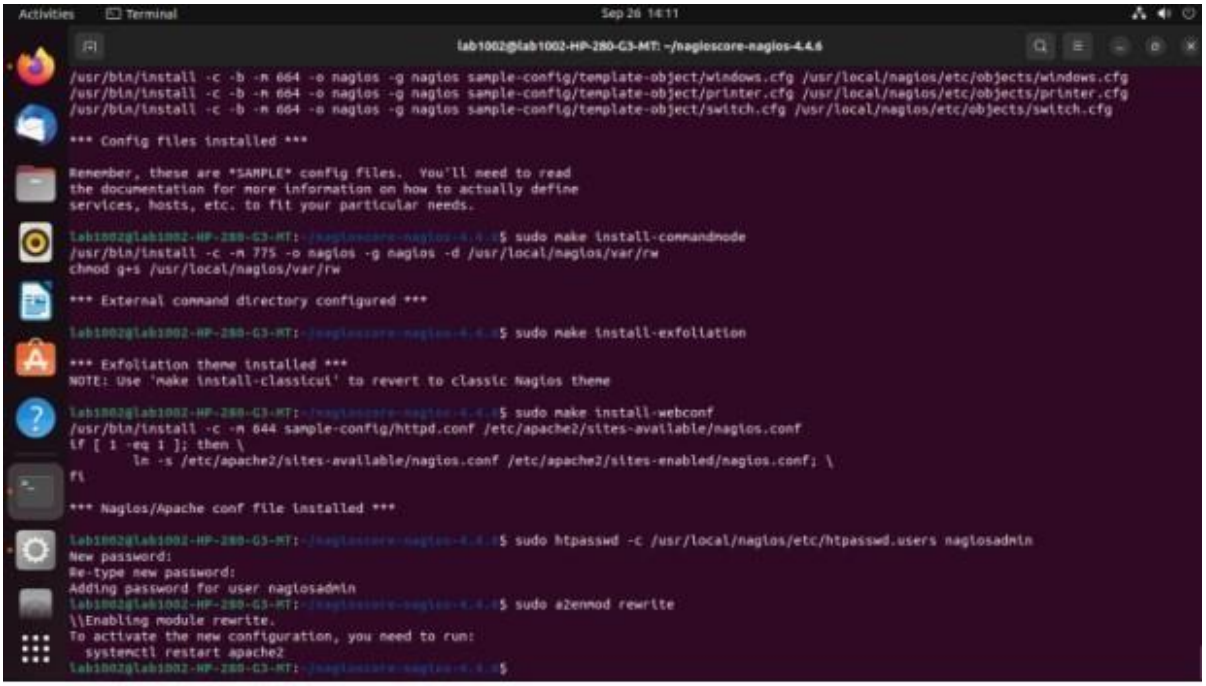

Activities
Terminal
Sep 26 14:02
lab1002@lab1002-HP-280-G3-MT: ~
Fetched 9,670 kB in 4s (2,328 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
201 packages can be upgraded. Run 'apt list --upgradable' to see them.
lab1002@lab1002-HP-280-G3-MT: $ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
firefox linux-headers-6.2.0-33-generic linux-hwe-6.2-headers-6.2.0-33 linux-image-6.2.0-33-generic linux-modules-6.2.0-33-generic
linux-modules-extra-6.2.0-33-generic
The following packages have been kept back:
gjs libgjs0g
The following packages will be upgraded:
alsa-ucm-conf apparmor apport apport-gtk apt apt-utils base-files bind9-dnsutils bind9-host bind9-libs cups cups-bsd cups-client
cups-common cups-core-drivers cups-daemon cups-ipp-utils cups-ppdc cups-server-common distro-info distro-info-data dpkg evince
evince-common file fwupd fwupd-signed gdm3 gir1.2-adw-1 gir1.2-gdm-1.0 gir1.2-gnomedesktop-3.0 gir1.2-gtk-4.0 gir1.2-nutter-10
gir1.2-pango-1.0 gnome-control-center gnome-control-center-data gnome-control-center-faces gnome-desktop3-data gnome-remote-desktop
gnome-session-canberra gnome-settings-daemon gnome-settings-daemon-common gnome-shell gnome-shell-common gnome-shell-extension-ubuntu-dock
im-config initramfs-tools initramfs-tools-bin initramfs-tools-core iptables isc-dhcp-client isc-dhcp-common libadwaita-1-0 libapparmor1
libapt-pkg6.0 libc-bin libc6 libc6-dbg libcanberra-gtk3-0 libcanberra-gtk3-module libcanberra-pulse libcanberra0 libcups2 libcupsimage2
libegl-mesa0 libevdocument3-4 libevview3-3 libflac8 libfprint-2-2 libfwupd2 libfwupdplugin5 libgbm1 libgdm1 libgl1-amber-dri
libgl1-mesa-dri libglapi-mesa libglx-mesa0 libgnome-bg-4-1 libgnome-desktop-3-19 libgnome-desktop-4-1 libgpgme11 libgpgmepp6
libgssapi-krb5-2 libgtk-4-1 libgtk-4-bin libgtk-4-common libidn12 libinput-bin libinput10 liblp4tc2 libip6tc2 libk5crypto3 libkrb5-3
libkrb5support0 libldap-2.5-0 libldap-common libmagic-mgc libmagic1 libnbin-glib4 libnbin-proxy libnm-glib0 libnutter-10-0
libnautilus-extension1a libnetplan0 libnss-systemd libpam-systemd libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpangoxft-1.0-0
libpcsclite1 libpulse-mainloop-glib0 libpulse0 libpulsedsp libqmi-glib5 libqmi-proxy libsasl2-2 libsasl2-modules libsasl2-modules-db
libsasl2-modules-gssapi-mit libsmbclient libsnmp-base libsnmp40 libspeechd2 libsystemd0 libudev1 libunwind8 libwbclient0 libwebp7
libwebpdemux2 libwebpmux3 libxatracker2 libxtables12 linux-firmware linux-generic-hwe-22.04 linux-headers-generic-hwe-22.04
linux-image-generic-hwe-22.04 locales mesa-vulkan-drivers modemmanager mokutil mutter-common nautilus nautilus-data netplan.io
openssh-client pulseaudio pulseaudio-module-bluetooth pulseaudio-utils python-apt-common python3-apport python3-apt python3-debian
python3-distro-info python3-distupgrade python3-macaroonbakery python3-problem-report python3-software-properties python3-speechd
python3-tz samba-libs software-properties-common software-properties-gtk speech-dispatcher speech-dispatcher-audio-plugins
speech-dispatcher-espeak-ng systemd systemd-hwe-hwdb systemd-oomd systemd-sysv systemd-timesyncd tcpdump thermald thunderbird
thunderbird-gnome-support thunderbird-locale-en thunderbird-locale-en-us tzdata ubuntu-advantage-tools ubuntu-desktop

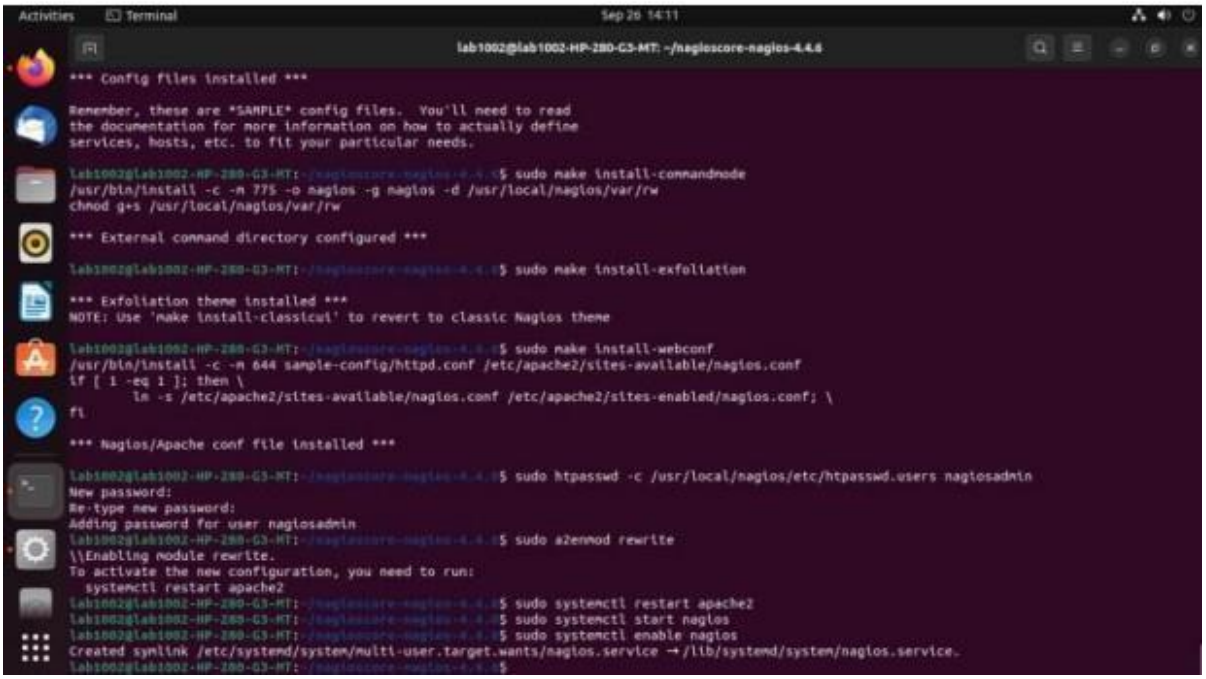

Activities
Terminal
Sep 26 14:05
lab1002@lab1002-HP-280-G3-MT: ~
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto mode
Setting up build-essential (12.9ubuntu3) ...
Setting up php (2:8.1+92ubuntu1) ...
Processing triggers for ufw (0.36.1-4ubuntu0.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.3) ...
Processing triggers for php8.1-cli (8.1.2-1ubuntu2.14) ...
Processing triggers for libapache2-mod-php8.1 (8.1.2-1ubuntu2.14) ...
lab1002@lab1002-HP-280-G3-MT: $ sudo useradd nagios
lab1002@lab1002-HP-280-G3-MT: $ sudo groupadd nagcmd
lab1002@lab1002-HP-280-G3-MT: $ sudo usermod -a -G nagcmd nagios
lab1002@lab1002-HP-280-G3-MT: $ sudo usermod -a -G nagcmd www-data
lab1002@lab1002-HP-280-G3-MT: $ wget https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
--2023-09-26 14:04:26-- https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/NagiosEnterprises/nagioscore/tar.gz/refs/tags/nagios-4.4.6 [following]
--2023-09-26 14:04:26-- https://codeload.github.com/NagiosEnterprises/nagioscore/tar.gz/refs/tags/nagios-4.4.6
Resolving codeload.github.com (codeload.github.com)... 20.207.73.88
Connecting to codeload.github.com (codeload.github.com)|20.207.73.88|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'nagios-4.4.6.tar.gz'
nagios-4.4.6.tar.gz [<=>] 10.81M 8.06MB/s in 1.3s
2023-09-26 14:04:28 (8.06 MB/s) - 'nagios-4.4.6.tar.gz' saved [11333431]
lab1002@lab1002-HP-280-G3-MT:~$ tar -zxvf nagios-4.4.6.tar.gz
nagioscore-nagios-4.4.6/
nagioscore-nagios-4.4.6/.gitignore
nagioscore-nagios-4.4.6/.travis.yml
nagioscore-nagios-4.4.6/CONTRIBUTING.md
nagioscore-nagios-4.4.6/Changelog
nagioscore-nagios-4.4.6/INSTALLING
nagioscore-nagios-4.4.6/LEGAL
nagioscore-nagios-4.4.6/LICENSE


Activities
Terminal
Sep 26 14:07
lab1002@lab1002-HP-280-G3-MT: ~/nagioscore-nagios-4.4.6
nagioscore-nagios-4.4.6/xdata/xodtemplate.c
nagioscore-nagios-4.4.6/xdata/xodtemplate.h
nagioscore-nagios-4.4.6/xdata/xpddefault.c
nagioscore-nagios-4.4.6/xdata/xpddefault.h
nagioscore-nagios-4.4.6/xdata/xrddefault.c
nagioscore-nagios-4.4.6/xdata/xrddefault.h
nagioscore-nagios-4.4.6/xdata/xsddefault.c
nagioscore-nagios-4.4.6/xdata/xsddefault.h
lab1002@lab1002-HP-280-G3-MT:~$ cd nagioscore -nagios-4.4.6
bash: cd: too many arguments
lab1002@lab1002-HP-280-G3-MT:~$ cd nagioscore-nagios-4.4.6
bash: ./configfure: No such file or directory
checking for a BSD-compatible install... /usr/bin/install -c
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking whether make sets $(MAKE)... yes
checking whether ln -s works... yes
checking for strip... /usr/bin/strip
checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for ANSI C header files... yes
checking whether time.h and sys/time.h may both be included... yes
checking for sys/wait.h that is POSIX.1 compatible... yes
checking for sys/types.h... yes
checking for sys/stat.h... yes
checking for stdlib.h... yes


Activities
Terminal
Sep 26 14:07
lab1002@lab1002-HP-280-G3-MT: ~/nagioscore-nagios-4.4.6
HTML URL: http://localhost/nagios/
CGI URL: http://localhost/nagios/cgi-bin/
Traceroute (used by WAP):
Review the options above for accuracy. If they look okay,
type 'make all' to compile the main program and CGIs.
$ make all
cd ./base && make
make[1]: Entering directory '/home/lab1002/nagioscore-nagios-4.4.6/base'
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nagios.o nagios.c
nagios.c: In function 'main':
nagios.c:611:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
nagios.c:841:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
nagios.c: In function 'nagios_core_worker':
nagios.c:176:17: warning: ignoring return value of 'read' declared with attribute 'warn_unused_result' [-Wunused-result]
nagios.c: In function 'test_path_access':
nagios.c:122:17: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o broker.o broker.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o nebmods.o nebmods.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o ../common/shared.o ../common/shared.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o query-handler.o query-handler.c
gcc -Wall -I.. -g -O2 -DHAVE_CONFIG_H -DNSCORE -c -o workers.o workers.c
workers.c: In function 'handle_worker_result':
workers.c:801:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]
workers.c:804:25: warning: ignoring return value of 'asprintf' declared with attribute 'warn_unused_result' [-Wunused-result]


Activities
Terminal
Sep 26 14:09
lab1002@lab1002-HP-280-G3-MT: ~/nagioscore-nagios-4.4.6
make[1]: Leaving directory '/home/lab1002/nagioscore-nagios-4.4.6'
$ sudo make install-init
/usr/bin/install -c -m 755 -d -o root -g root /lib/systemd/system
/usr/bin/install -c -m 755 -o root -g root startup/default-service /lib/systemd/system/nagios.service
$ sudo make install-config
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/etc/objects
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/nagios.cfg /usr/local/nagios/etc/nagios.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/cgi.cfg /usr/local/nagios/etc/cgi.cfg
/usr/bin/install -c -b -m 660 -o nagios -g nagios sample-config/resource.cfg /usr/local/nagios/etc/resource.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/templates.cfg /usr/local/nagios/etc/objects/templates.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/commands.cfg /usr/local/nagios/etc/objects/commands.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/contacts.cfg /usr/local/nagios/etc/objects/contacts.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/timeperiods.cfg /usr/local/nagios/etc/objects/timeperiods.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/localhost.cfg /usr/local/nagios/etc/objects/localhost.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg
*** Config files installed ***
Remember, these are *SAMPLE* config files. You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.
$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
*** External command directory configured ***
$ sudo make install-exfoliation
*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme


Activities
Terminal
Sep 26 14:11
lab1002@lab1002-HP-280-G3-MT: ~/nagioscore-nagios-4.4.6
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/windows.cfg /usr/local/nagios/etc/objects/windows.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/printer.cfg /usr/local/nagios/etc/objects/printer.cfg
/usr/bin/install -c -b -m 664 -o nagios -g nagios sample-config/template-object/switch.cfg /usr/local/nagios/etc/objects/switch.cfg
*** Config files installed ***
Remember, these are *SAMPLE* config files. You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
*** External command directory configured ***
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-exfoliation
*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-available/nagios.conf
if [1 -eq 1]; then \
ln -s /etc/apache2/sites-available/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \
fi
*** Nagios/Apache conf file installed ***
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo a2enmod rewrite
\\Enabling module rewrite.
To activate the new configuration, you need to run:
systemctl restart apache2
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$


Activities
Terminal
Sep 26 14:11
lab1002@lab1002-HP-280-G3-MT: ~/nagioscore-nagios-4.4.6
*** Config files installed ***
Remember, these are *SAMPLE* config files. You'll need to read
the documentation for more information on how to actually define
services, hosts, etc. to fit your particular needs.
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-commandmode
/usr/bin/install -c -m 775 -o nagios -g nagios -d /usr/local/nagios/var/rw
chmod g+s /usr/local/nagios/var/rw
*** External command directory configured ***
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-exfoliation
*** Exfoliation theme installed ***
NOTE: Use 'make install-classicui' to revert to classic Nagios theme
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo make install-webconf
/usr/bin/install -c -m 644 sample-config/httpd.conf /etc/apache2/sites-available/nagios.conf
if [1 -eq 1]; then \
ln -s /etc/apache2/sites-available/nagios.conf /etc/apache2/sites-enabled/nagios.conf; \
fi
*** Nagios/Apache conf file installed ***
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
New password:
Re-type new password:
Adding password for user nagiosadmin
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo a2enmod rewrite
\\Enabling module rewrite.
To activate the new configuration, you need to run:
systemctl restart apache2
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo systemctl restart apache2
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo systemctl start nagios
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$ sudo systemctl enable nagios
Created symlink /etc/systemd/system/multi-user.target.wants/nagios.service → /lib/systemd/system/nagios.service.
lab1002@lab1002-HP-280-G3-MT:~/nagioscore-nagios-4.4.6$

CONCLUSION:

In this assignment, we successfully installed Nagios on an Ubuntu system, setting up a robust monitoring and alerting framework for efficient system management and proactive issue resolution.