Name: Andaya, Lyka C.	**Date Performed: October 16, 2023**
Course/Section: CPE31S4	**Date Submitted: October 16, 2023**
Instructor: Dr. Taylar	**Semester and SY: 2023-2024**

Activity 8: Install, Configure, and Manage Availability Monitoring tools

1. Objectives

Create and design a workflow that installs, configure and manage enterprise monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

2. Discussion

Availability monitoring is a type of monitoring tool that we use if the certain workload is up or reachable on our end. Site downtime can lead to loss of revenue, reputational damage and severe distress. Availability monitoring prevents adverse situations by checking the uptime of infrastructure components such as servers and apps and notifying the webmaster of problems before they impact on business.

3. Tasks

1. Create a playbook that installs Nagios in both Ubuntu and CentOS. Apply the concept of creating roles.
2. Describe how you did step 1. (Provide screenshots and explanations in your report. Make your report detailed such that it will look like a manual.)
3. Show an output of the installed Nagios for both Ubuntu and CentOS.
4. Make sure to create a new repository in GitHub for this activity.

4. Output (screenshots and explanations

INPUT	lykaandaya@managenode: ~/HOA8.1
	GNU nano 6.2 install.yml
	`---`
	`- hosts: all`
	`  become: true`
	`  tasks:`
	
	`  - name: update package cache for Ubuntu`
	`  apt:`
	`  update_cache: yes`
	`  when: ansible_distribution == "Ubuntu"`
	
	`  - name: update package cache for CentOS`
	`  dnf:`
	`  name: epel-release`
	`  state: latest`
	`  when: ansible_distribution == "CentOS"`
	
	`- hosts: nagios`
	`  become: true`
	`  roles:`
	`  - nagios`

lykaandaya@managenode: ~/HOA8.1/roles/nagios/tasks

GNU nano 6.2 main.yml

```
- name: install nagios fot Ubuntu
  package:
    name:
      - nagios4
      - nagios-plugins
  when: ansible_distribution == "Ubuntu"

- name: enable and start nagios service
  service:
    name: nagios4
    state: started
    enabled: yes
  when: ansible_distribution == "Ubuntu"

- name: install nagios for CentOS
  dnf:
    name:
      - nagios
      - nagios-plugins-all
  when: ansible_distribution == "CentOS"

- name: enable and start nagios service
  service:
    name: nagios
    state: started
    enabled: yes
  when: ansible_distribution == "CentOS"
```

Explanation: In this playbook it will install the nagios4 on Ubuntu server 2 with the use of apt package manager and also on CentOS but with the use of dnf package manager. It also ensures that the Nagios service is enabled and started on both Ubuntu and CentOS.

Process	

```
lykaandaya@managenode:~/HOA8.1$ ansible-playbook --ask-become-pass install.yml
BECOME password:

PLAY [all] *********************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.113]
ok: [192.168.56.110]
ok: [192.168.56.114]

TASK [update package cache for Ubuntu] *****************************************
skipping: [192.168.56.110]
changed: [192.168.56.113]
changed: [192.168.56.114]

TASK [update package cache for CentOS] *****************************************
skipping: [192.168.56.113]
skipping: [192.168.56.114]
ok: [192.168.56.110]

PLAY [nagios] ******************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.113]
ok: [192.168.56.110]

TASK [nagios : install nagios fot Ubuntu] **************************************
skipping: [192.168.56.110]
ok: [192.168.56.113]

TASK [nagios : enable and start nagios service] ********************************
skipping: [192.168.56.110]
ok: [192.168.56.113]

TASK [nagios : install nagios for CentOS] **************************************
skipping: [192.168.56.113]
ok: [192.168.56.110]
```

```
TASK [nagios : enable and start nagios service] ********************************
skipping: [192.168.56.113]
ok: [192.168.56.110]

PLAY RECAP *********************************************************************
192.168.56.110             : ok=5    changed=0    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.113             : ok=5    changed=1    unreachable=0    failed=0    s
kipped=3    rescued=0    ignored=0
192.168.56.114             : ok=2    changed=1    unreachable=0    failed=0    s
kipped=1    rescued=0    ignored=0
```

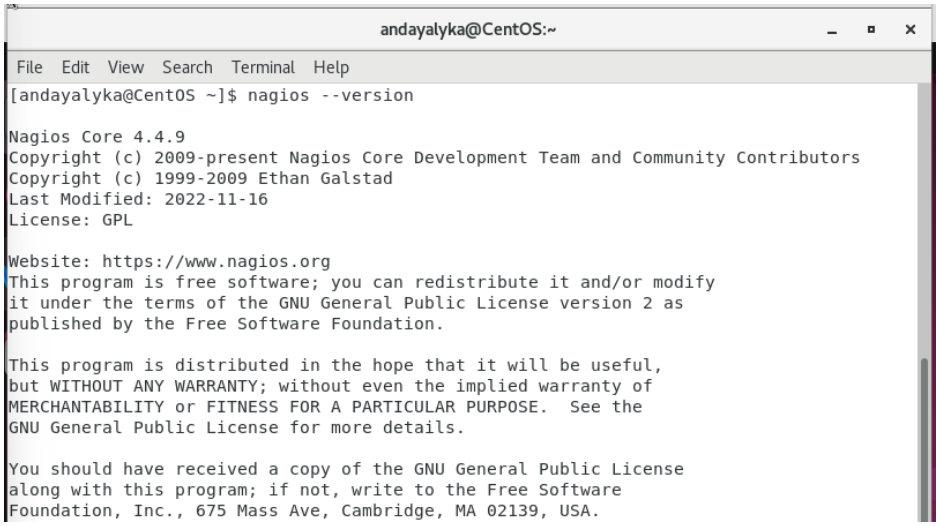
Explanation: It shows that it executed the instructions in the tasks of the playbook I created.

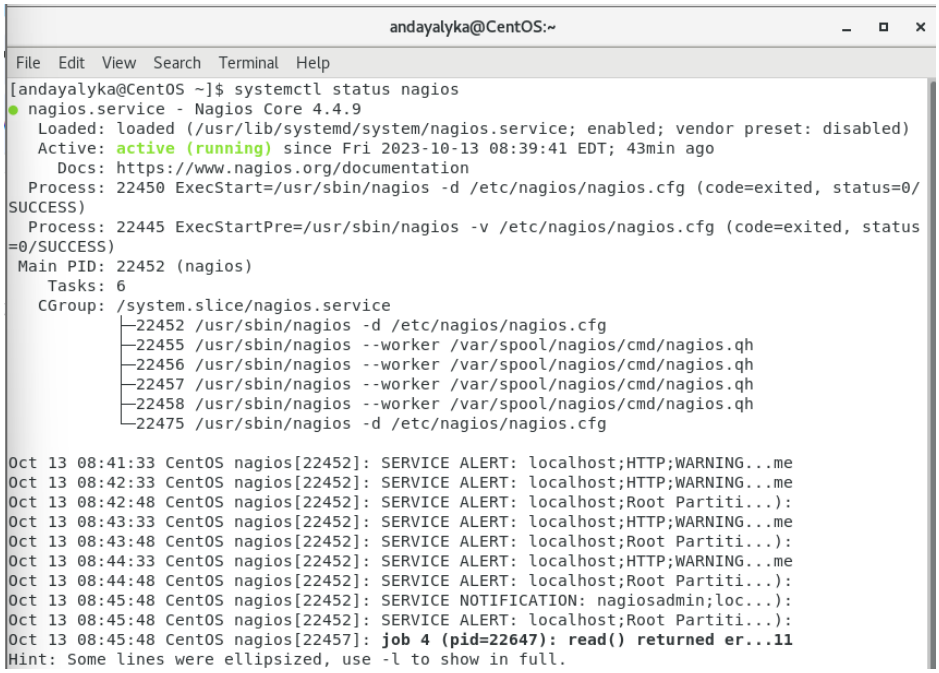
Output	
Ubuntu	

```
lykaandaya@controlnode2:~$ nagios4 --version

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Commu
nity Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org
Files rogram is free software; you can redistribute it and/or mod
ify
it under the terms of the GNU General Public License version 2 as
published by the Free Software Foundation.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

lykaandaya@controlnode2:~$ systemctl status nagios4
● nagios4.service - nagios4
     Loaded: loaded (/lib/systemd/system/nagios4.service; enable>
     Active: active (running) since Mon 2023-10-16 11:35:58 PST;>
       Docs: man:nagios4
   Main PID: 32642 (nagios4)
      Tasks: 6 (limit: 1054)
     Memory: 5.8M
        CPU: 444ms
     CGroup: /system.slice/nagios4.service
             ├─32642 /usr/sbin/nagios4 /etc/nagios4/nagios.cfg
             ├─32643 /usr/sbin/nagios4 --worker /var/lib/nagios4>
             ├─32644 /usr/sbin/nagios4 --worker /var/lib/nagios4>
             ├─32645 /usr/sbin/nagios4 --worker /var/lib/nagios4>
             ├─32646 /usr/sbin/nagios4 --worker /var/lib/nagios4>
             └─32731 /usr/sbin/nagios4 /etc/nagios4/nagios.cfg
```

```
Oct 16 11:35:58 controlnode2 nagios4[32642]: wproc: Registry req>
Oct 16 11:35:58 controlnode2 nagios4[32642]: wproc: Registry req>
Oct 16 11:35:58 controlnode2 nagios4[32642]: wproc: Registry req>
Oct 16 11:35:59 controlnode2 nagios4[32642]: Successfully launch>
Oct 16 11:35:59 controlnode2 nagios4[32642]: Successfully launch>
Oct 16 11:39:05 controlnode2 nagios4[32642]: SERVICE ALERT: loca>
Oct 16 11:40:05 controlnode2 nagios4[32642]: SERVICE ALERT: loca>
Oct 16 11:41:05 controlnode2 nagios4[32642]: SERVICE ALERT: loca>
Oct 16 11:42:05 controlnode2 nagios4[32642]: SERVICE NOTIFICATIO>
Oct 16 11:42:05 controlnode2 nagios4[32642]: SERVICE ALERT: loca>
lines 1-26/26 (END)
```

CentOS





Explanation: In Ubuntu, it indicates that nagios4 is installed and the service is currently active and running. Meanwhile, in CentOS, nagios is installed and its service is also active and running.

```
lykaandaya@managenode:~/HOA8.1$ git add *
lykaandaya@managenode:~/HOA8.1$ git commit -m "HOA8"
[main 2804862] HOA8
 3 files changed, 57 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 install.yml
 create mode 100644 inventory
lykaandaya@managenode:~/HOA8.1$ git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 810 bytes | 810.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:andayalyka/HOA8.1.git
   f37763b..2804862  main -> main
```

Explanation: It shows that it added the 3 files and it has been pushed in the github

Reflections:

Answer the following:

1. What are the benefits of having an availability monitoring tool?
   - **Availability monitoring tools provide ongoing checks on the status of services, applications, and systems. This ensures timely identification of problems or potential failures, allowing for swift response and reducing downtime. Possessing such a tool is a crucial element in upholding a dependable and effective IT framework, a necessity for the prosperity and competitiveness of contemporary enterprises.**

Conclusions:

Developing and architecting a workflow for the installation, configuration, and administration of enterprise-grade monitoring tools using Ansible necessitates a fusion of advanced Ansible proficiency, in-depth familiarity with the intricacies of monitoring solutions, adeptness in Infrastructure as Code (IaC) principles, and the application of industry-standard methodologies in orchestrating workflows. Moreover, it hinges on a robust grounding in IT operations and expert-level systems administration.