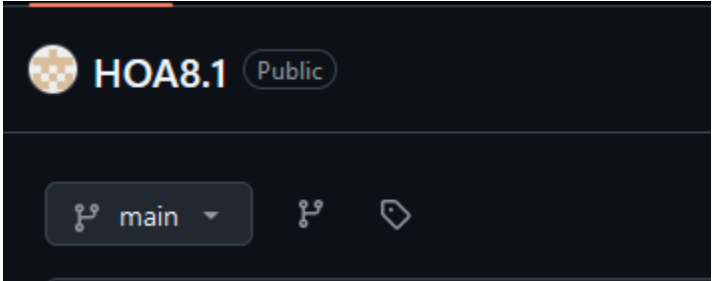


Name: Jessie Robert Lazo	Date Performed: 10/14/2024
Course/Section: CPE 212-CPE31S2	Date Submitted: 10/14/2024
Instructor: Engr. Robin Valenzuela	Semester and SY:
Activity 8: Install, Configure, and Manage Availability Monitoring tools	
1. Objectives	
Create and design a workflow that installs, configure 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	
<ol style="list-style-type: none"> 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)	
<p>Before beginning the activity, the first thing to do is create a new repository, creating the ansible configuration file and inventory file needed to establish a working ansible environment between one local machine (ubuntu), and two remote machines (Ubuntu, and CentOS).</p>	
	

```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help
GNU nano 2.9.3 inventory

[remote_servers]
192.168.56.101 ansible_python_interpreter=/usr/bin/python3
192.168.56.109 ansible_python_interpreter=/usr/bin/python
```

```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help
GNU nano 2.9.3 ansible.cfg

[defaults]

inventory = inventory
host_key_checking = False

deprecation_warning = False

remote_user = jessielazo
private_key_file = ~/.ssh/
```

2. Create a playbook named nagios.yml and implement basic update and upgrade commands for Ubuntu and CentOS machines.

```
jessielazo@Desktop: ~/NOA8
File Edit View Search Terminal Help
GNU nano 2.9.3 nagios.yml Mi
---
- hosts: all
  become: true
  pre_tasks:
    - name: install update and repositories (CentOS)
      tags: always
      yum:
        name: "*"
        update_cache: yes
        state: latest
      changed_when: false
      when: ansible_distribution == "CentOS"
    - name: install update and repositories (Ubuntu)
      tags: always
      apts:
        upgrade: yes
        update_cache: yes
        cache_valid_time: 86400
      changed_when: false
      when: ansible_distribution == "Ubuntu"
```

3. In setting up the Nagios, there are a lot of dependencies which has to be installed first for they would be used later, during, or upon installation of Nagios.

```
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: Installing nagios dependencies and libraries (CentOS)
  tags: dependencies, libraries
  yum:
    name:
      - gcc
      - glibc
      - glibc-common
      - perl
      - httpd
      - php
      - wget
      - gd
      - gd-devel
      - openssl-devel
      - gcc
      - glibc
      - glibc-common
      - make
      - gettext
      - automake
      - autoconf
      - wget
      - openssl-devel

[ Read 106 lines ]
```

These are the list of dependencies which consists of libraries, services, and stuff needed for the setup of Nagios.

4. Upon installing the required dependencies for setting up Nagios, the next thing to do is to create a directory for storing the Nagios files and plugins which would later be downloaded.

jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help

GNU nano 2.9.3

main.yml

```
- net-snmp-utils
- python3-pip
  state: present
  when: ansible_distribution == "CentOS"

- name: Install passlib python package (CentOS)
  pip:
    executable: /usr/bin/pip3
    name: passlib
    when: ansible_distribution == "CentOS"

- name: Creating a directory for nagios (CentOS)
  file:
    path: ~/nagios
    state: directory
    when: ansible_distribution == "CentOS"
```

jessielazo@Desktop: ~/HOA8/roles/remote_servers_ubuntu/tasks
File Edit View Search Terminal Help

GNU nano 2.9.3

main.yml

```
- python3-pip
  state: latest
  when: ansible_distribution == "Ubuntu"

- name: passlib package (Ubuntu)
  pip:
    name: passlib
    when: ansible_distribution == "Ubuntu"

- name: Creating a directory for nagios (Ubuntu)
  file:
    path: ~/nagios
    state: directory
    when: ansible_distribution == "Ubuntu"
```

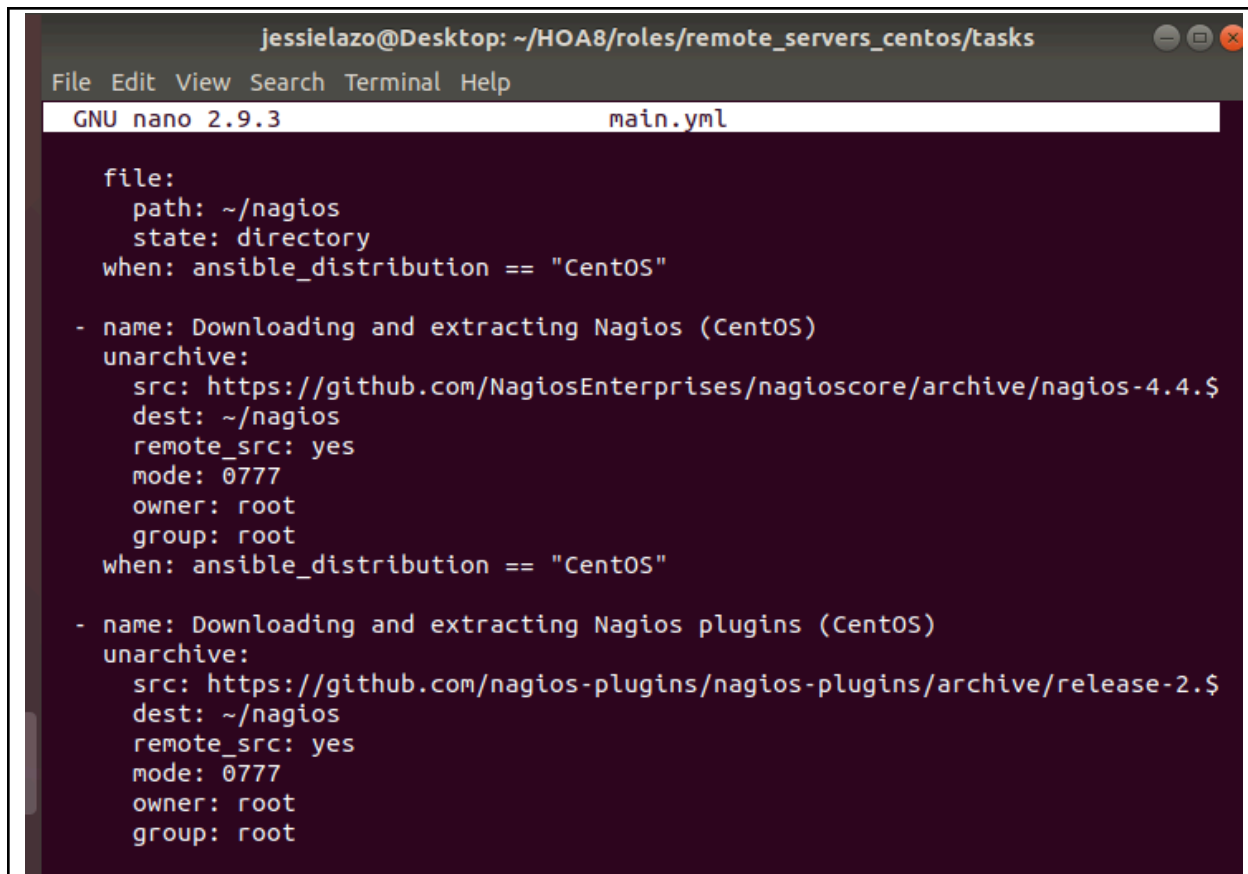
5. After creating a directory, we can start downloading and extracting Nagios files and Plugins

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_ubuntu/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

    state: directory
  when: ansible_distribution == "Ubuntu"

- name: Downloading and extracting Nagios (Ubuntu)
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root
  when: ansible_distribution == "Ubuntu"

- name: Downloading and extracting Nagios plugins (Ubuntu)
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root
  when: ansible_distribution == "Ubuntu"
```



```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

file:
  path: ~/nagios
  state: directory
when: ansible_distribution == "CentOS"

- name: Downloading and extracting Nagios (CentOS)
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root
  when: ansible_distribution == "CentOS"

- name: Downloading and extracting Nagios plugins (CentOS)
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root
```

6. When Nagios files and Plugins are already extracted, we can begin the installation and additional procedures which are the following shell commands as part of the setting up of Nagios.

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: Compiling, installing, and adding users and groups in nagios (CentOS)
  shell: |
    cd ~/nagios/nagioscore-**
    ./configure
    make all
    make install-groups-users
    usermod -a -G nagios apache
    make install
    make install-daemoninit
    make install-commandmode
    make install-config
    make install-webconf
  when: ansible_distribution == "CentOS"

- name: Compiling and installing plugins (CentOS)
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install
  when: ansible_distribution == "CentOS"
```

```
- name: install, compile, adding users and groups in Nagios (Ubuntu)
  shell: |
    cd ~/nagios/nagioscore-*
    sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
    sudo make all
    sudo make install-groups-users
    sudo usermod -a -G nagios www-data
    sudo make install
    sudo make install-daemoninit
    sudo make install-commandmode
    sudo make install-config
    sudo make install-webconf
```



```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_ubuntu/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

shell: |
  cd ~/nagios/nagioscore-*
  sudo ./configure --with-httpd-conf=/etc/apache2/sites-enabled
  sudo make all
  sudo make install-groups-users
  sudo usermod -a -G nagios www-data
  sudo make install
  sudo make install-daemoninit
  sudo make install-commandmode
  sudo make install-config
  sudo make install-webconf
  sudo a2enmod rewrite
  sudo a2enmod cgi
when: ansible_distribution == "Ubuntu"

- name: compile and install plugins (Ubuntu)
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install
  when: ansible_distribution == "Ubuntu"
```

This includes compiling and installing Nagios files and plugins, and lastly adding a user and its permissions

7. After running the necessary installation and configurations of Nagios files and plugins, we have to make sure the Nagios and apache/httpd services are in UP state.

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_ubuntu/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: Add a user and permissions (Ubuntu)
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: chrysler
    password: chrysler
  when: ansible_distribution == "Ubuntu"

- name: starting/enabling nagios (Ubuntu)
  service:
    name: nagios
    state: restarted
    enabled: true
  when: ansible_distribution == "Ubuntu"

- name: starting/enabling apache2 (Ubuntu)
  service:
    name: apache2
    state: restarted
    enabled: true
  when: ansible_distribution == "Ubuntu"
```

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help
GNU nano 2.9.3 main.yml

- name: Add a user and permissions (CentOS)
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: chrysler
    password: chrysler
  when: ansible_distribution == "CentOS"
  vars:
    ansible_python_interpreter: /usr/bin/python3

- name: starting/enabling nagios (CentOS)
  service:
    name: nagios
    state: restarted
    enabled: true
  when: ansible_distribution == "CentOS"

- name: starting/enabling apache2 (CentOS)
  service:
    name: httpd
    state: restarted
    enabled: true
  when: ansible_distribution == "CentOS"
```

8. To make the management more convenient, it is highly suggested to implement roles. Make a directory named roles, then make its required subdirectories for implementing the role

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help
jessielazo@Desktop:~/HOA8$ ls
ansible.cfg  inventory  nagios.yml  roles
jessielazo@Desktop:~/HOA8$ cd roles
jessielazo@Desktop:~/HOA8/roles$ mkdir remote_servers_centos
jessielazo@Desktop:~/HOA8/roles$ mkdir remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles$ ls
remote_servers_centos  remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles$ cd remote_servers_centos
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ ls
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ sudo nano main.yml
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ cd ..
jessielazo@Desktop:~/HOA8/roles$ ls
remote_servers_centos  remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles$ cd remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles/remote_servers_ubuntu$ cd ..
jessielazo@Desktop:~/HOA8/roles$ ls
remote_servers_centos  remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles$ cd remote_servers_centos
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ ls
main.yml
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ ls
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ mkdir tasks
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ cd tasks
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos/tasks$ sudo nano main.yml
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos/tasks$ cd ..
jessielazo@Desktop:~/HOA8/roles/remote_servers_centos$ cd ..
jessielazo@Desktop:~/HOA8/roles$ cd remote_servers_ubuntu
jessielazo@Desktop:~/HOA8/roles/remote_servers_ubuntu$ mkdir tasks
jessielazo@Desktop:~/HOA8/roles/remote_servers_ubuntu$ cd tasks
```

9. Modify the nagios.yml file according to how the roles in a playbook are implemented

```
---
- hosts: all
  become: true
  pre_tasks:

  - name: install update and repositories (CentOS)
    tags: always
    yum:
      name: "*"
      update_cache: yes
      state: latest
    changed_when: false
    when: ansible_distribution == "CentOS"

  - name: install update and repositories (Ubuntu)
    tags: always
    apts:
      upgrade: yes
      update_cache: yes
      cache_valid_time: 86400
    changed_when: false
    when: ansible_distribution == "Ubuntu"
```

```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help
GNU nano 2.9.3 nagios.yml

  when: ansible_distribution == "CentOS"

- name: install update and repositories (Ubuntu)
  tags: always
  apts:
    upgrade: yes
    update_cache: yes
    cache_valid_time: 86400
  changed_when: false
  when: ansible_distribution == "Ubuntu"

- hosts: all
  become: true
  roles:
    - base

- hosts: remote_servers
  become: true
  roles:
    - remote_servers_ubuntu
    - remote_servers_centos
```

10. The main.yml file for ubuntu_servers and centos_servers must be copied from the old nagios.yml file.

jessielazo@Desktop: ~/HOA8/roles/remote_servers_ubuntu/tasks

File Edit View Search Terminal Help

GNU nano 2.9.3

main.yml

```
- name: nagios libraries and dependencies (Ubuntu)
  tags: ubuntu, dependencies, libraries
  apt:
    name:
      - autoconf
      - libc6
      - gcc
      - make
      - wget
      - unzip
      - apache2
      - php
      - libapache2-mod-php
      - libgd-dev
      - openssl
      - libssl-dev
      - bc
      - gawk
      - dc
      - build-essential
      - snmp
      - libnet-snmp-perl
      - gettext
```

```
jessielazo@Desktop: ~/HOA8/roles/remote_servers_centos/tasks
File Edit View Search Terminal Help
```

```
GNU nano 2.9.3
```

```
main.yml
```

```
- name: Installing nagios dependencies and libraries (CentOS)
  tags: dependencies, libraries
  yum:
    name:
      - gcc
      - glibc
      - glibc-common
      - perl
      - httpd
      - php
      - wget
      - gd
      - gd-devel
      - openssl-devel
      - gcc
      - glibc
      - glibc-common
      - make
      - gettext
      - automake
      - autoconf
      - wget
      - openssl-devel
```

11. Run the nagios.yml and check and debug for errors, and lastly provide proof for working Nagios.

12. Sync the repository with Github.

Outputs:

1st run: Ubuntu runs successfully without error. I still encounter centos errors like in previous activity.


```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help

TASK [install update and repositories (Ubuntu)] *****
*
[WARNING]: Could not find aptitude. Using apt-get instead.
ok: [192.168.56.101]

PLAY [all] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.101]

PLAY [remote_servers] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.101]

TASK [remote_servers_ubuntu : nagios libraries and dependencies (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : passlib package (Ubuntu)] *****
*
changed: [192.168.56.101]
```

```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help
TASK [remote_servers_ubuntu : nagios libraries and dependencies (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : passlib package (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : Creating a directory for nagios (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : Downloading and extracting Nagios plugins (Ubuntu)] ***
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : install, compile, adding users and groups in Nagios (Ubuntu)] ***
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : compile and install plugins (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : starting/enabling nagios (Ubuntu)] *****
```

```
jessielazo@Desktop: ~/HOA8
File Edit View Search Terminal Help

TASK [remote_servers_ubuntu : Downloading and extracting Nagios plugins (Ubuntu
)] ***
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : install, compile, adding users and groups in Nagi
os (Ubuntu)] ***
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : compile and install plugins (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : starting/enabling nagios (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_ubuntu : starting/enabling apache2 (Ubuntu)] *****
*
changed: [192.168.56.101]

TASK [remote_servers_centos : Installing nagios dependencies and libraries (Cent
OS)] ***
skipping: [192.168.56.101]

TASK [remote_servers_centos : Install passlib python package (CentOS)] *****
*
skipping: [192.168.56.101]

PLAY RECAP *****
*
192.168.56.101      : ok=13   changed=9   unreachable=0   failed=0
192.168.56.109      : ok=1    changed=0   unreachable=0   failed=1
jessielazo@Desktop:~/HOA8$
```

Nagios®

General

Home

Documentation

Current Status

Tactical Overview

Map (Legacy)

Hosts

Services

Host Groups

Summary

Grid

Service Groups

Summary

Grid

Problems

Services (Unhandled)

Hosts (Unhandled)

Network Outages

Quick Search:

Reports

Availability

Trends (Legacy)

Nagios® Core™

✓ Daemon running with PID 827

Nagios® Core™

Version 4.4.6

April 28, 2020

[Check for updates](#)

A new version of Nagios Core is available!

Visit nagios.org to download Nagios 4.5.6.

Get Started

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

Quick Links

- [Nagios Library](#) (tutorials and docs)
- [Nagios Labs](#) (development blog)
- [Nagios Exchange](#) (plugins and addons)
- [Nagios Support](#) (tech support)
- [Nagios.com](#) (company)
- [Nagios.org](#) (project)

Page Tour

Reflections:

Answer the following:

1. What are the benefits of having an availability monitoring tool?

Being able to use an availability monitoring tool is like being a networking administrator on top of all your remote machines which you can easily see which specific areas have a problem such as what services. This is what the Nagios do in an enterprise, it eases the monitoring work between remote machines especially in a large enterprise.

Conclusions:

This activity primarily focused on giving the experience of what it is like to install, configure, and manage enterprise availability monitoring through the ansible. The very objective of the activity is to create and design a workflow that installs, configures, and manages enterprise monitoring tools using Ansible as

an Infrastructure as Code (IaC) tool. The activity does not provide steps on what should be done to accomplish the objective hence I have read references on how to set up the Nagios including the prerequisites for installing the Nagios such as its dependencies and plugins. In setting up the Nagios, we were tasked to descriptively explain each and every step we have done to accomplish the activity so I made my own procedure. The procedures were divided into three groups, first is the installation of Nagios prerequisites which includes the creation of a new repository, an inventory file, and an ansible configuration file given that the activity started from scratch. To talk about the prerequisites of Nagios, one of these are the dependencies needed before beginning its setup which includes a lot of libraries and services. Next is the creation of a downloaded files directory for the Nagios files and then downloading the necessary files and plugins before starting the actual installation. Moving forward to the next task which begins the setup, the installation of Nagios files and plugins was performed at this point including the necessary compilations of specific files. Also, the creation of a user needed for the Nagios (and its permissions) was performed in this part. Then the last task was to implement roles to further organize the use of ansible playbook. I made the necessary folders in order to be able to implement the roles, and then I did some debugging, checkings, and lastly the proof of installation. Upon I ended performing the activity, I had successfully achieved my intended learning outcomes for this activity, and I was able to get the key takeaways from this activity.