| | |
|---|---|
| **Name:** Repani, Justin Jello J. | **Date Performed:** October 9, 2023 |
| **Course/Section:** CPE31S6 - CPE232 | **Date Submitted:** October 9, 2023 |
| **Instructor:** Dr. Jonathan V. Taylar | **Semester and SY:** 1st Sem - 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**

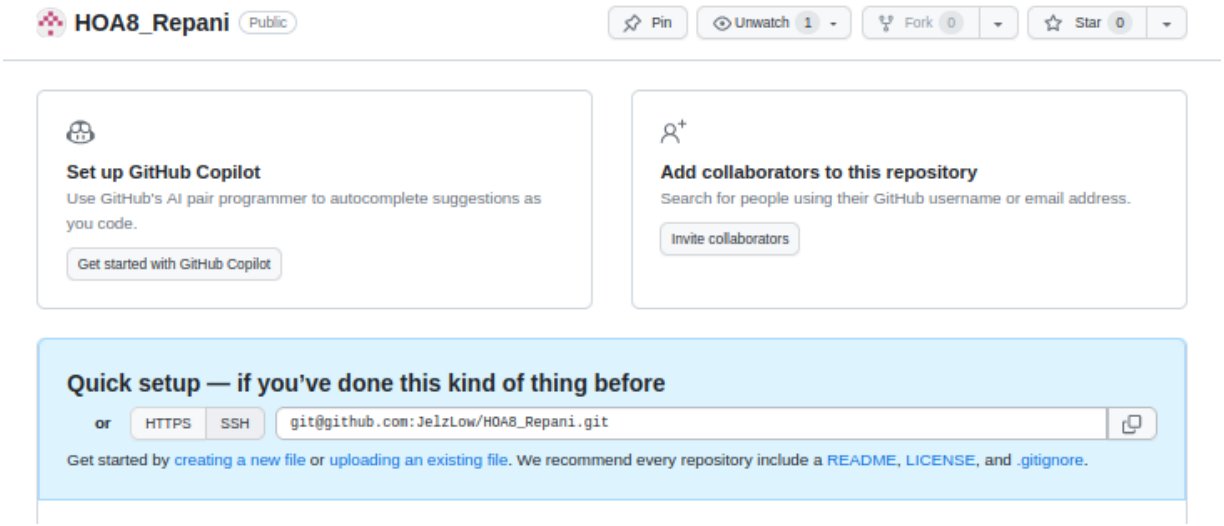
Availability monitoring is a type of monitoring tool that we use if the certain workload is up or reachable on our end. Site downtime can lead to loss of revenue, reputational damage and severe distress. Availability monitoring prevents adverse situations by checking the uptime of infrastructure components 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.

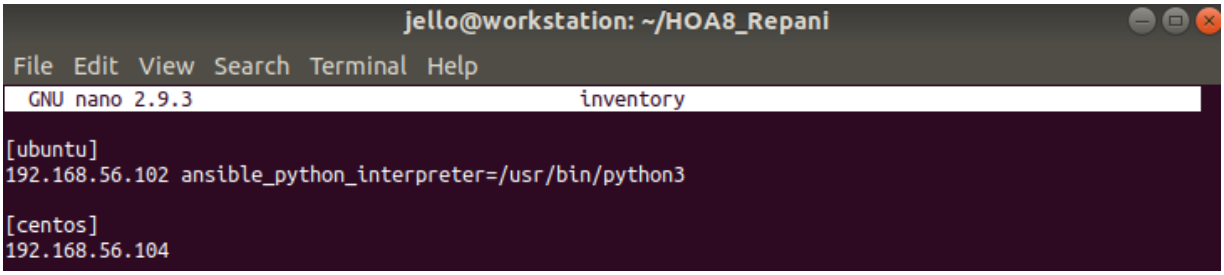
**4. Output** (screenshots and explanations)

Create a new repository in Git and use the command git clone in order to clone the repository to the workstation machine




```
jello@workstation:~$ git clone git@github.com:JelzLow/HOA8_Repani.git
Cloning into 'HOA8_Repani'...
warning: You appear to have cloned an empty repository.
jello@workstation:~$ ls
CPE232_hoa6     Documents          HOA7_Repani  Pictures           Templates
CPE232_Repani   Downloads          HOA8_Repani  Public             token.txt
Desktop         examples.desktop   Music        Repani_PrelimExam  Videos
```

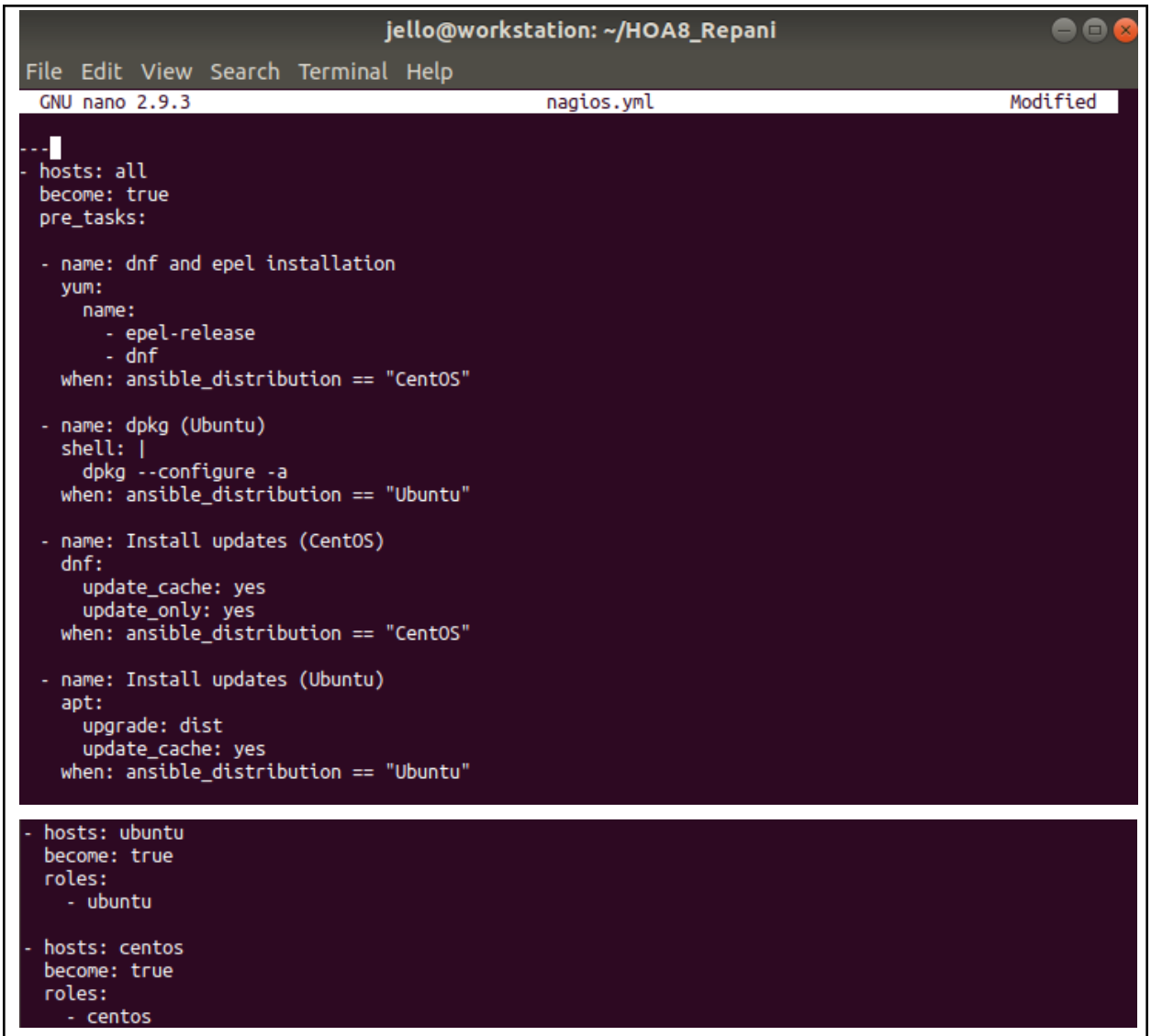
Create an inventory file which contains the IP addresses of the remote servers along with the appropriate groups. There are two in this case, one is for the Ubuntu server with the group name [ubuntu], and the other is the CentOS server with the group name [centos]



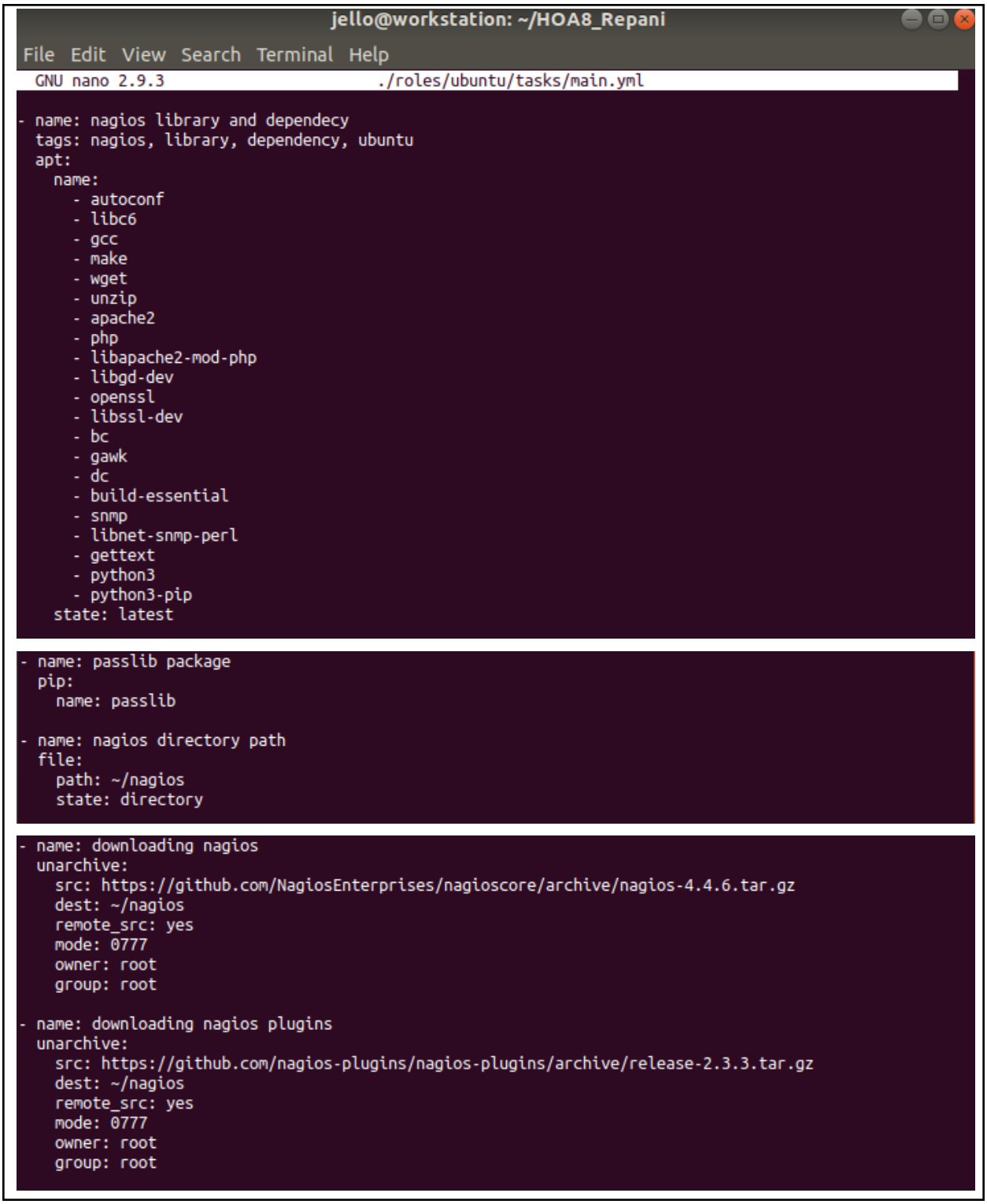
Create a roles directory containing two directories “centos” and “ubuntu” each with its own tasks directory inside

```
jello@workstation:~/HOA8_Repani$ sudo nano inventory
jello@workstation:~/HOA8_Repani$ mkdir roles
jello@workstation:~/HOA8_Repani$ cd roles
jello@workstation:~/HOA8_Repani/roles$ mkdir ubuntu centos
jello@workstation:~/HOA8_Repani/roles$ mkdir ubuntu/tasks
jello@workstation:~/HOA8_Repani/roles$ mkdir centos/tasks
jello@workstation:~/HOA8_Repani/roles$ cd ..
jello@workstation:~/HOA8_Repani$ tree
.
├── ansible.cfg
├── inventory
└── roles
    ├── centos
    │   └── tasks
    └── ubuntu
        └── tasks

5 directories, 2 files
```

Create a file called “nagios.yml” this will contain the playbook commands that will perform the pre tasks and call on to the main tasks inside the roles/tasks directories.



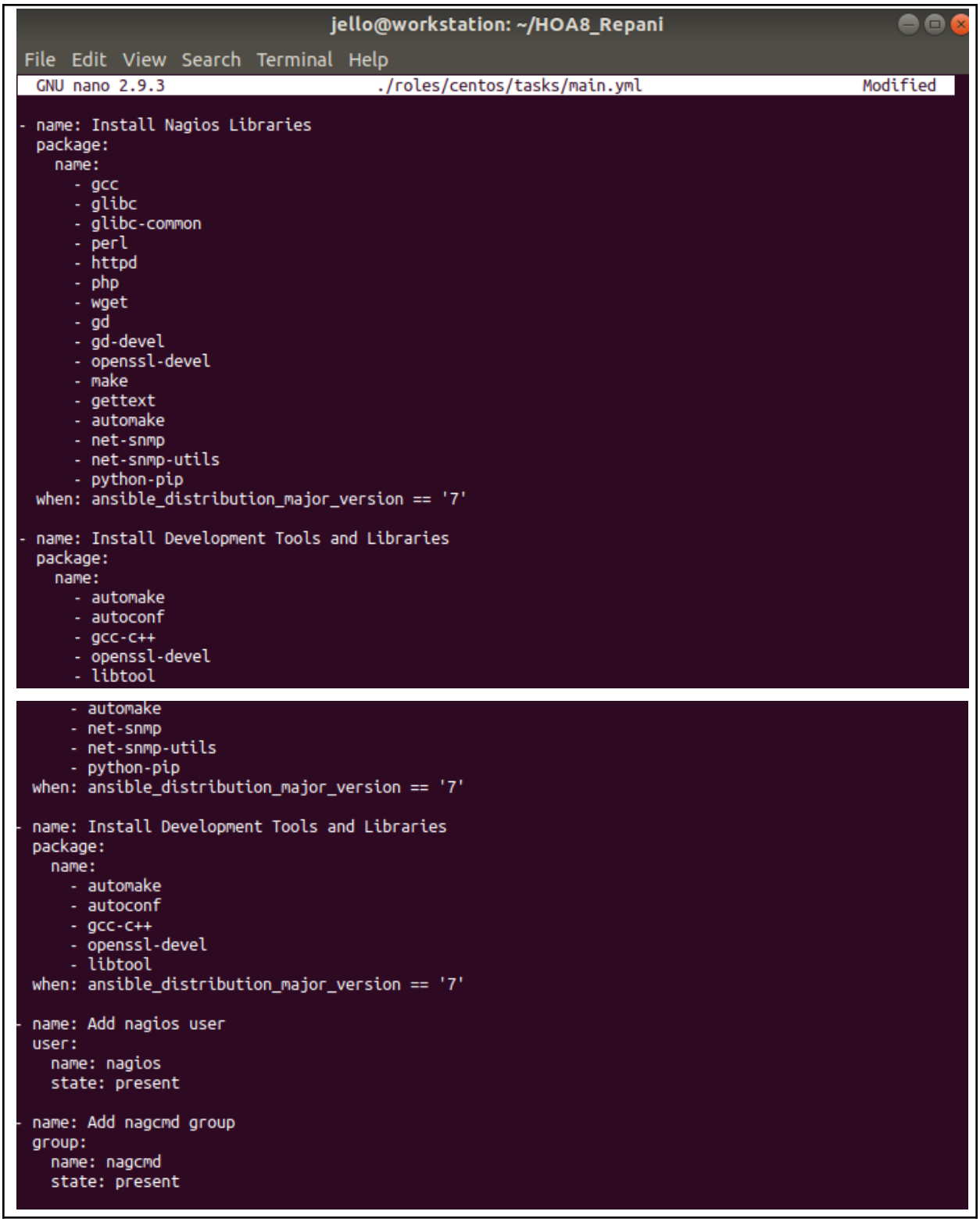

Create a main.yml file inside the tasks directory of ubuntu in roles. This will run the script necessary to download and run nagios in Ubuntu. The playbook contains the function to download all the libraries required to run nagios, after this we add a nagios user before installing and configuring nagios.


jello@workstation: ~/HOA8_Repani
File Edit View Search Terminal Help
GNU nano 2.9.3 ./roles/ubuntu/tasks/main.yml
- name: nagios library and dependecy
tags: nagios, library, dependency, ubuntu
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
- python3
- python3-pip
state: latest
- name: passlib package
pip:
name: passlib
- name: nagios directory path
file:
path: ~/nagios
state: directory
- name: downloading nagios
unarchive:
src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
dest: ~/nagios
remote_src: yes
mode: 0777
owner: root
group: root
- name: downloading nagios plugins
unarchive:
src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.3.tar.gz
dest: ~/nagios
remote_src: yes
mode: 0777
owner: root
group: root

```
- name: install, compile, adding users and groups
  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
```

```
- name: compile and install plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install

- name: adding users to nagios
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin

- name: Nagios Start/Enable Check
  service:
    name: nagios
    state: restarted
    enabled: true
```

Create a main.yml file inside the tasks directory of centos in roles. This will run the script necessary to download and run nagios in CentOS. The playbook contains the function to download all the libraries required to run nagios, after this we add a nagios user before installing and configuring nagios.


jello@workstation: ~/HOA8_Repani
File Edit View Search Terminal Help
GNU nano 2.9.3 ./roles/centos/tasks/main.yml Modified
- name: Install Nagios Libraries
package:
name:
- gcc
- glibc
- glibc-common
- perl
- httpd
- php
- wget
- gd
- gd-devel
- openssl-devel
- make
- gettext
- automake
- net-snmp
- net-snmp-utils
- python-pip
when: ansible_distribution_major_version == '7'
- name: Install Development Tools and Libraries
package:
name:
- automake
- autoconf
- gcc-c++
- openssl-devel
- libtool
- automake
- net-snmp
- net-snmp-utils
- python-pip
when: ansible_distribution_major_version == '7'
- name: Install Development Tools and Libraries
package:
name:
- automake
- autoconf
- gcc-c++
- openssl-devel
- libtool
when: ansible_distribution_major_version == '7'
- name: Add nagios user
user:
name: nagios
state: present
- name: Add nagcmd group
group:
name: nagcmd
state: present

```
- name: Add nagios to nagcmd group
  user:
    name: nagios
    groups: nagcmd
    append: yes

- name: Add apache to nagcmd group
  user:
    name: apache
    groups: nagcmd
    append: yes

- name: Create Nagios directory PATH
  file:
    path: ~/nagios
    state: directory
```

```
- name: Download Nagios
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.tar.gz
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Download Nagios plugins
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.3.tar.gz
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
```

```
    group: root

- name: Configure Nagios
  command: >
    ./configure --with-command-group=nagcmd
  args:
    chdir: ~/nagios
```

```
- name: Compile and Install Nagios plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
    ./configure
    make
    make install

- name: Install Nagios Web Configuration
  shell: make install-webconf
  args:
    chdir: ~/nagios

- name: Adding Users to Nagios
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin

- name: Start Apache
  service:
    name: httpd
    state: started
    enabled: yes
```

```
- name: Change directory to Nagios installation directory
  command: cd ~/nagios

- name: Verify Nagios Configuration
  command: >
    /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
  changed_when: false

- name: Add Nagios service to startup
  systemd:
    name: nagios.service
    enabled: yes
  when: ansible_distribution_major_version == '7'
```

```
- name: Start Nagios service
  service:
    name: nagios
    state: started
  when: ansible_distribution_major_version == '7'
```

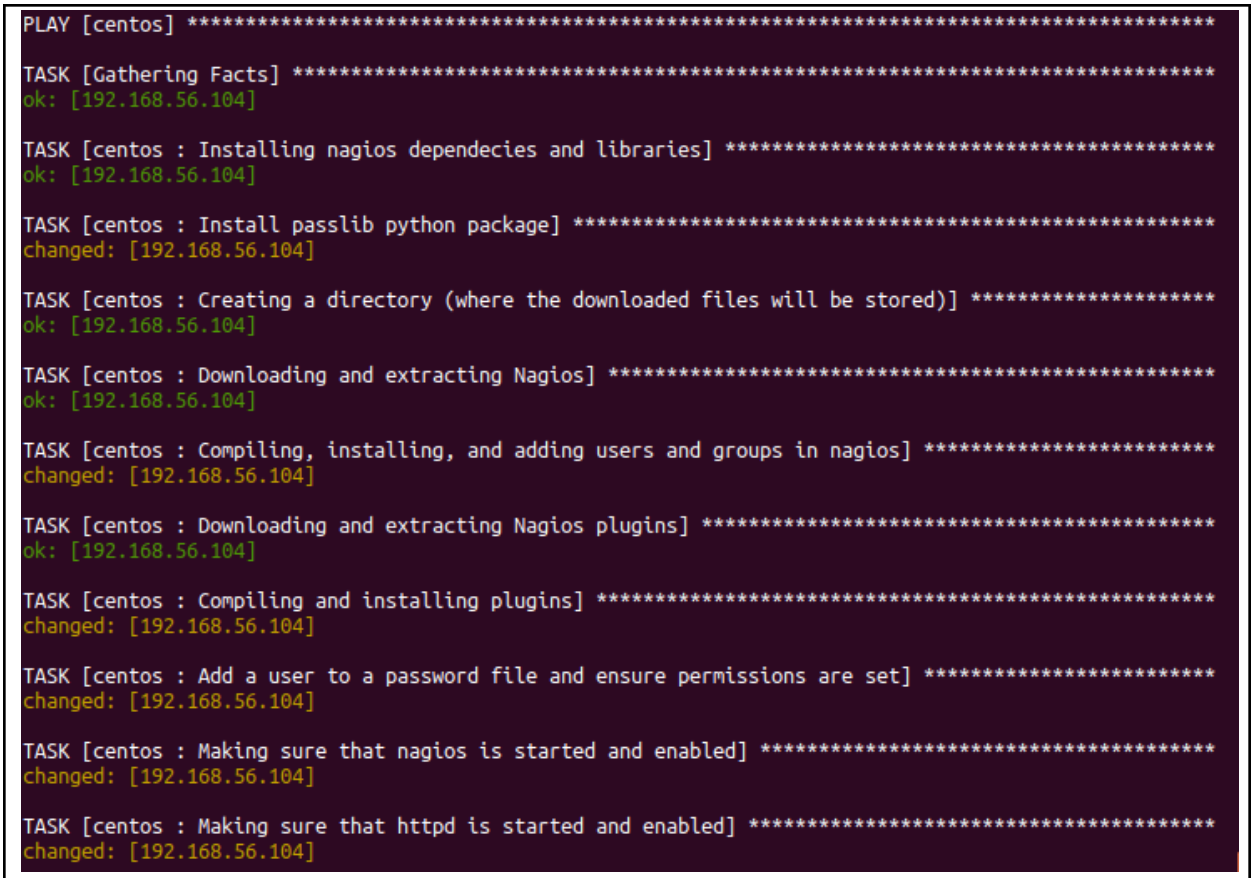
After performing the creation of files and directories, the repository should now look like this.

```
jello@workstation:~/HOA8_Repani$ tree
.
├── ansible.cfg
├── inventory
├── nagios.yml
└── roles
    ├── centos
    │   └── tasks
    │       └── main.yml
    └── ubuntu
        └── tasks
            └── main.yml

5 directories, 5 files
```

Running the playbook

```
jello@workstation:~/HOA8_Repani$ ansible-playbook --ask-become-pass nagios.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]
ok: [192.168.56.104]

TASK [dnf and epel installation] ***********************************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]

TASK [dpkg (Ubuntu)] ***********************************************************************
skipping: [192.168.56.104]
changed: [192.168.56.102]

TASK [Install updates (CentOS)] ************************************************************
skipping: [192.168.56.102]
ok: [192.168.56.104]

TASK [Install updates (Ubuntu)] ************************************************************
skipping: [192.168.56.104]
ok: [192.168.56.102]
```

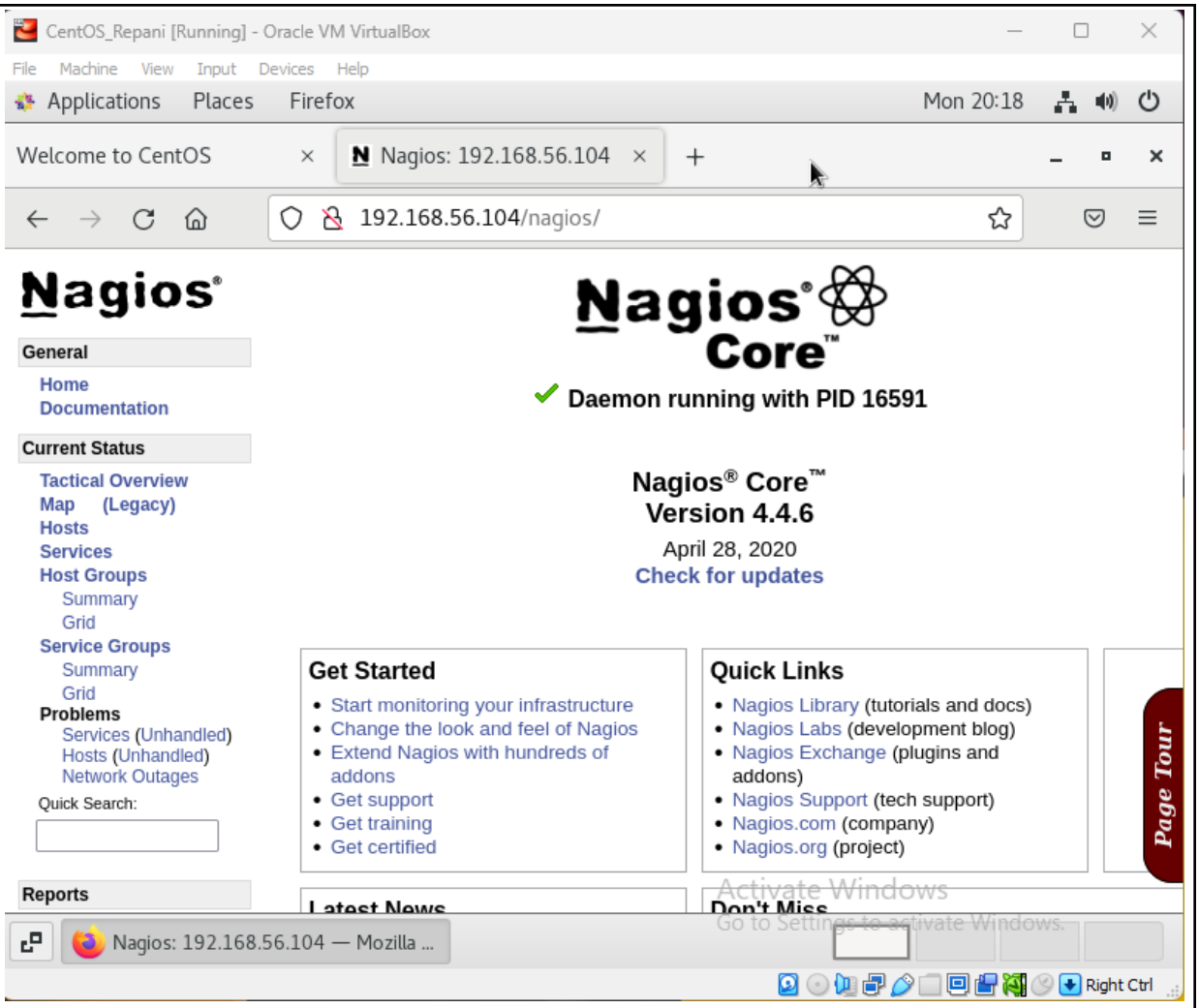
```
PLAY [ubuntu] ******************************************************************************

TASK [Gathering Facts] *********************************************************************
ok: [192.168.56.102]

TASK [ubuntu : nagios library and dependecy] ***********************************************
ok: [192.168.56.102]

TASK [ubuntu : passlib package] ************************************************************
ok: [192.168.56.102]

TASK [ubuntu : nagios directory path] ******************************************************
ok: [192.168.56.102]

TASK [ubuntu : downloading nagios] *********************************************************
ok: [192.168.56.102]

TASK [ubuntu : downloading nagios plugins] *************************************************
ok: [192.168.56.102]

TASK [ubuntu : install, compile, adding users and groups] **********************************
changed: [192.168.56.102]

TASK [ubuntu : compile and install plugins] ************************************************
changed: [192.168.56.102]

TASK [ubuntu : adding users to nagios] *****************************************************
ok: [192.168.56.102]

TASK [ubuntu : Nagios Start/Enable Check] **************************************************
changed: [192.168.56.102]
```




Proof of Nagios. In order to check if Nagios has been successfully downloaded, we must enter the ip address of the device and add /nagios at the end of the link to open

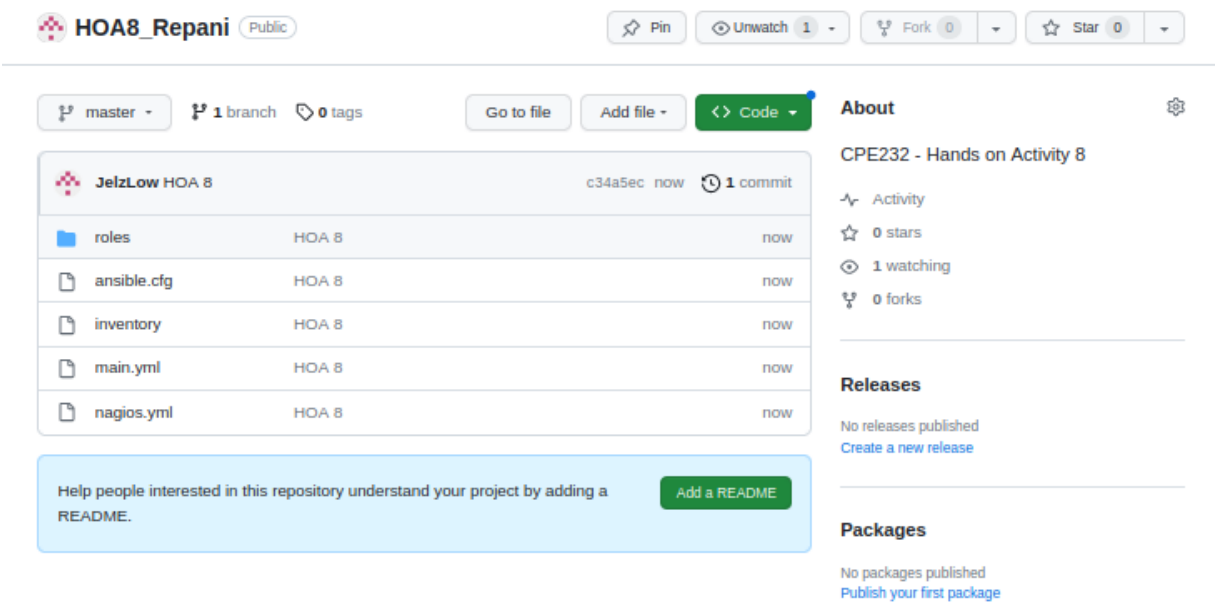
the website.


Server1_Repani [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
Mon 18:53
Nagios: 192.168.56.102
192.168.56.102/nagios/
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 29194
Nagios® Core™
Version 4.4.6
April 28, 2020
Check for updates
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)
Latest News
Don't Miss...
Page Tour
Activate Windows
Right Ctrl




Sync local machine with the Git Repository

```
jello@workstation:~/HOA8_Repani$ git add *
jello@workstation:~/HOA8_Repani$ git commit -m "HOA 8"
[master (root-commit) c34a5ec] HOA 8
 6 files changed, 421 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 inventory
 create mode 100644 main.yml
 create mode 100644 nagios.yml
 create mode 100644 roles/centos/tasks/main.yml
 create mode 100644 roles/ubuntu/tasks/main.yml
jello@workstation:~/HOA8_Repani$ git push origin
Counting objects: 13, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (13/13), 3.15 KiB | 1.57 MiB/s, done.
Total 13 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To github.com:JelzLow/HOA8_Repani.git
 * [new branch]      master -> master
jello@workstation:~/HOA8_Repani$
```




https://github.com/JelzLow/HOA8_Repani

**Reflections:**

Answer the following:

1. What are the benefits of having an availability monitoring tool?
   - The benefits of having a monitoring tools is that you can help administrators to manage the remote servers more effectively and efficiently since it takes logs of all the things that happen within the system and help notify the user about this information.

**Conclusions:**

Hands-on Activity 8 is about the installation, configuration, and managing of availability monitoring tools. In this activity the task is to install specifically in Ubuntu and CentOS while using the creation of roles. There aren't any given codes and syntax and instead we have to make our own playbook necessary and present all the steps taken in order to accomplish the task. This task is very challenging but thankfully we were allowed to collaborate with our seatmates which made the experience more bearable.

**Honor Pledge:**

*"I affirm that I have not given or received any unauthorized help on this assignment, and that this work is my own."*