| | |
|---|---|
| **Name:** Ian Carlo T. Bello | **Date Performed:** October 24, 2022 |
| **Course/Section:** CPE31S24 | **Date Submitted:** October 24, 2022 |
| **Instructor:** Dr. Jonathan V. Taylar | **Semester and SY:** 1st sem – 3rd year |

## 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

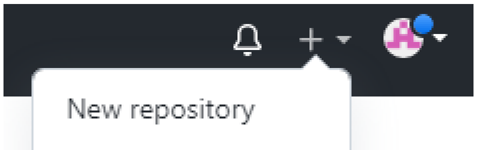
Availability monitoring is a type of monitoring tool that we use if the certain workload is up or reachable on our end. Site downtime can lead to loss of revenue, reputational damage and severe distress. Availability monitoring prevents adverse situations by checking the uptime of infrastructure components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

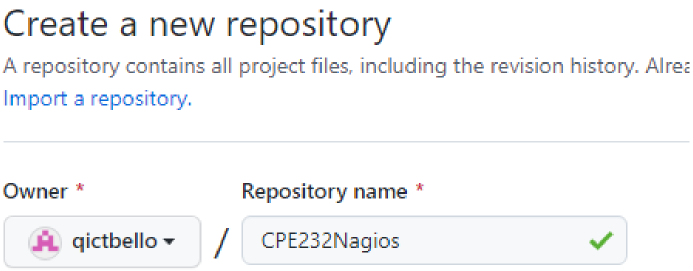
## 3. Tasks

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)

First, we need to create a new repository in our GitHub



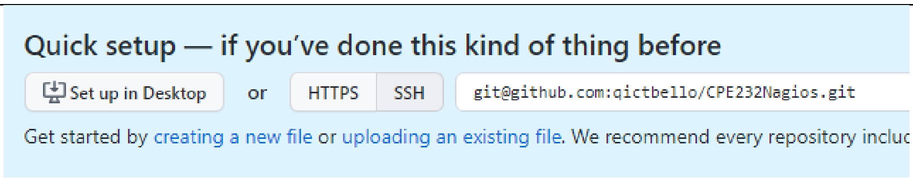


Create a new repository

A repository contains all project files, including the revision history. Alrea

Import a repository.

Owner * qictbello / Repository name * CPE232Nagios

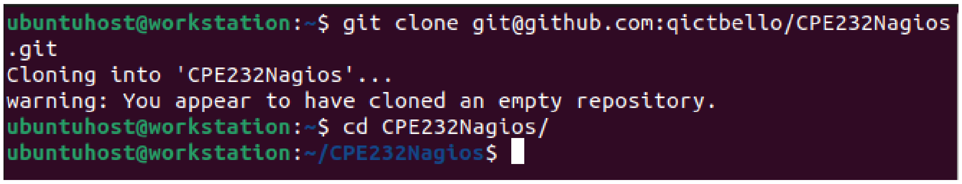


Quick setup — if you've done this kind of thing before

git@github.com:qictbello/CPE232Nagios.git

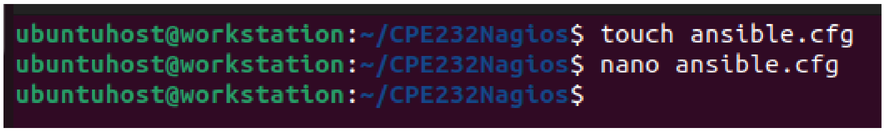
Get started by creating a new file or uploading an existing file. We recommend every repository incluc

We will name it CPE232Nagios since we are installing nagios in both server.

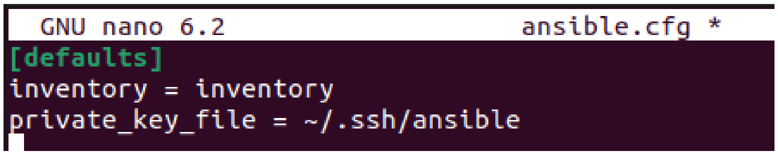


```
ubuntuhost@workstation:~$ git clone git@github.com:qictbello/CPE232Nagios.git
Cloning into 'CPE232Nagios'...
warning: You appear to have cloned an empty repository.
ubuntuhost@workstation:~$ cd CPE232Nagios/
ubuntuhost@workstation:~/CPE232Nagios$
```

We need to git clone it in our control node and work on its directory.

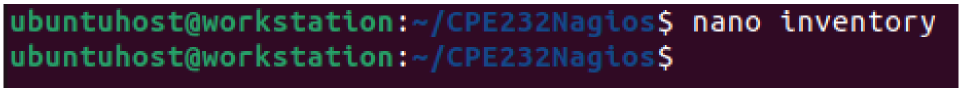


```
ubuntuhost@workstation:~/CPE232Nagios$ touch ansible.cfg
ubuntuhost@workstation:~/CPE232Nagios$ nano ansible.cfg
ubuntuhost@workstation:~/CPE232Nagios$
```

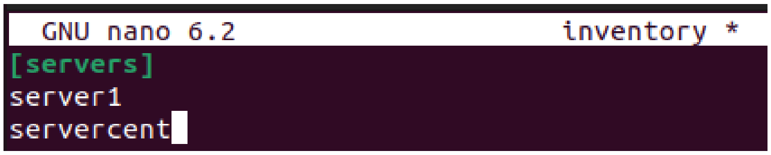


```
  GNU nano 6.2                      ansible.cfg *
[defaults]
inventory = inventory
private_key_file = ~/.ssh/ansible
```

Here we created ansible.cfg and next we will create inventory for the ip of servers.

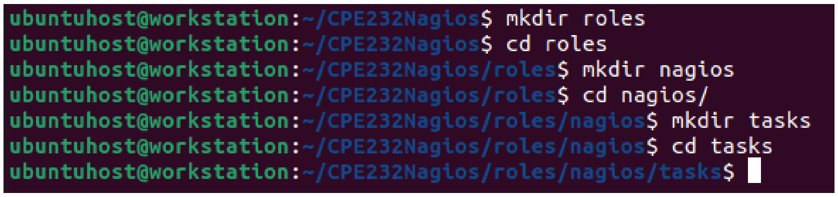


```
ubuntuhost@workstation:~/CPE232Nagios$ nano inventory
ubuntuhost@workstation:~/CPE232Nagios$
```



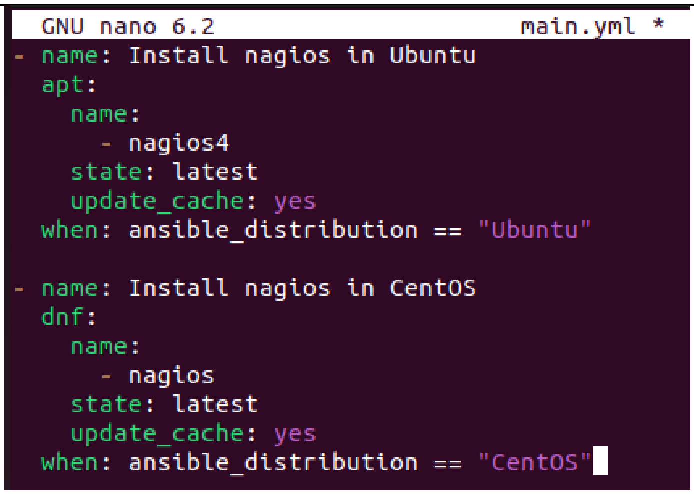
```
  GNU nano 6.2                        inventory *
[servers]
server1
servercent
```

Next we will be creating yml playbook to run and the role for installing nagios. We can search up how to install nagios and nagios4 are for ubuntu and nagios is for centos.



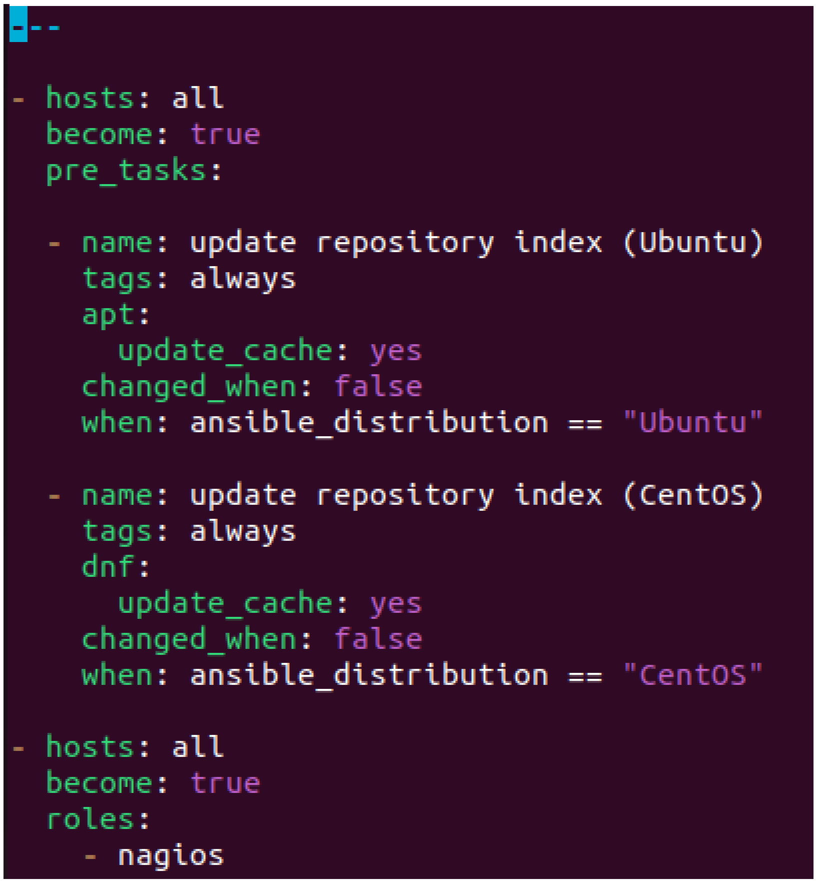
```
ubuntuhost@workstation:~/CPE232Nagios$ mkdir roles
ubuntuhost@workstation:~/CPE232Nagios$ cd roles
ubuntuhost@workstation:~/CPE232Nagios/roles$ mkdir nagios
ubuntuhost@workstation:~/CPE232Nagios/roles$ cd nagios/
ubuntuhost@workstation:~/CPE232Nagios/roles/nagios$ mkdir tasks
ubuntuhost@workstation:~/CPE232Nagios/roles/nagios$ cd tasks
ubuntuhost@workstation:~/CPE232Nagios/roles/nagios/tasks$
```

We created a directory for roles>nagios>tasks here we will create the main command in installing nagios for both servers.



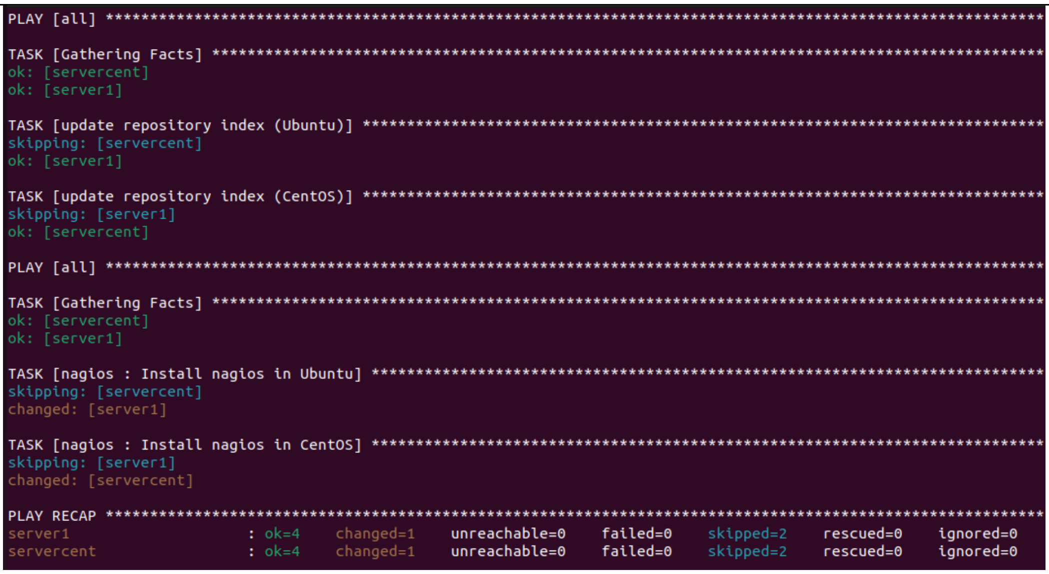
```
- name: Install nagios in Ubuntu
  apt:
    name:
      - nagios4
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: Install nagios in CentOS
  dnf:
    name:
      - nagios
    state: latest
    update_cache: yes
  when: ansible_distribution == "CentOS"
```

Next, we need to go back into our directory and create our yml playbook we will name it nagios.yml



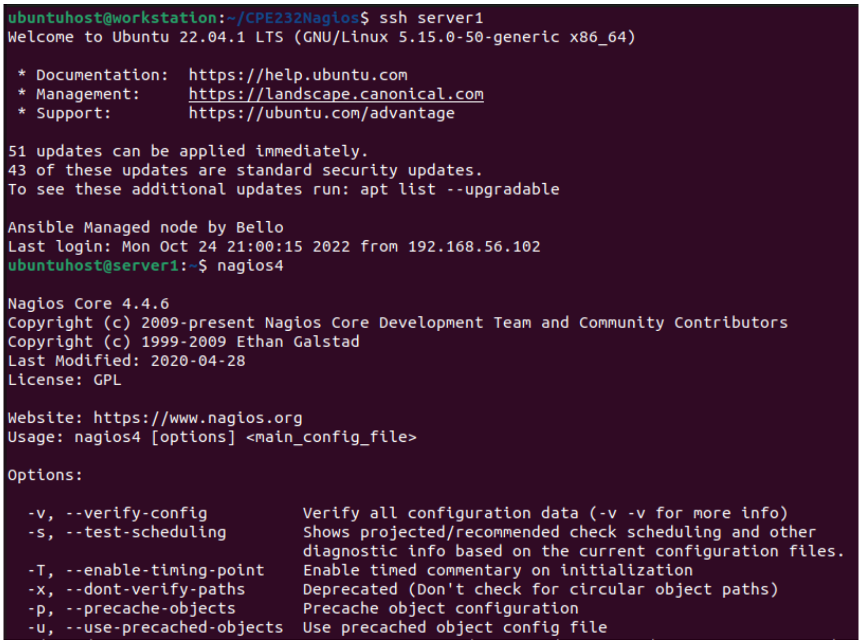
```
---

- hosts: all
  become: true
  pre_tasks:

  - name: update repository index (Ubuntu)
    tags: always
    apt:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "Ubuntu"

  - name: update repository index (CentOS)
    tags: always
    dnf:
      update_cache: yes
    changed_when: false
    when: ansible_distribution == "CentOS"

- hosts: all
  become: true
  roles:
    - nagios
```

As you can see we checked for updates and we run the roles nagios this will install nagios in both ubuntu and centos. We will run it now using ansible-playbook –ask-become-pass nagios.yml

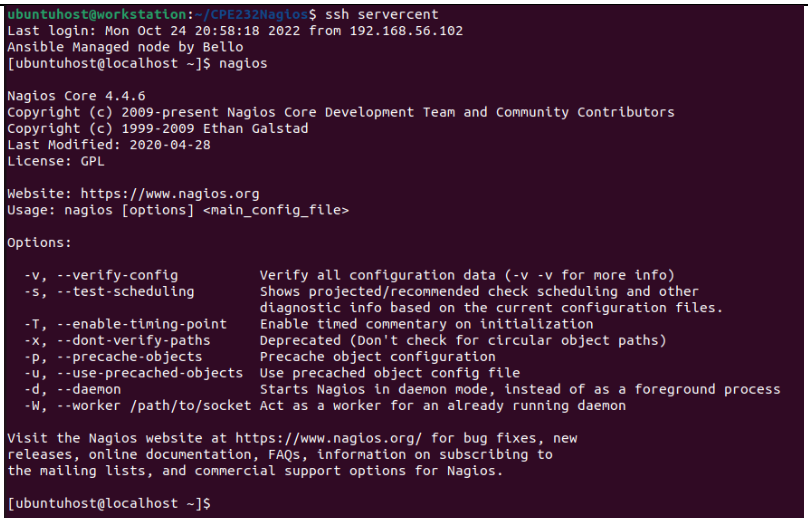


```
PLAY [all] *****************************************************************************************************

TASK [Gathering Facts] *****************************************************************************************
ok: [servercent]
ok: [server1]

TASK [update repository index (Ubuntu)] ************************************************************************
skipping: [servercent]
ok: [server1]

TASK [update repository index (CentOS)] ************************************************************************
skipping: [server1]
ok: [servercent]

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

TASK [Gathering Facts] *****************************************************************************************
ok: [servercent]
ok: [server1]

TASK [nagios : Install nagios in Ubuntu] ***********************************************************************
skipping: [servercent]
changed: [server1]

TASK [nagios : Install nagios in CentOS] ***********************************************************************
skipping: [server1]
changed: [servercent]

PLAY RECAP *****************************************************************************************************
server1                    : ok=4    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
servercent                 : ok=4    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
```

We can see that nagios are both installed and made change for both servers. Now we will check if both server do have it.

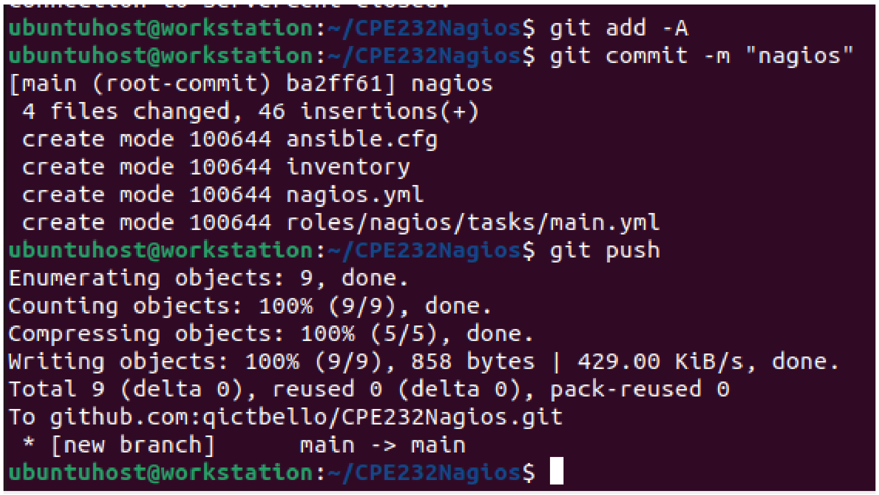


```
ubuntuhost@workstation:~/CPE232Nagios$ ssh server1
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-50-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

51 updates can be applied immediately.
43 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Ansible Managed node by Bello
Last login: Mon Oct 24 21:00:15 2022 from 192.168.56.102
ubuntuhost@server1:~$ nagios4

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

Website: https://www.nagios.org
Usage: nagios4 [options] <main_config_file>

Options:

  -v, --verify-config          Verify all configuration data (-v -v for more info)
  -s, --test-scheduling        Shows projected/recommended check scheduling and other
                               diagnostic info based on the current configuration files.
  -T, --enable-timing-point    Enable timed commentary on initialization
  -x, --dont-verify-paths      Deprecated (Don't check for circular object paths)
  -p, --precache-objects       Precache object configuration
  -u, --use-precached-objects  Use precached object config file
```



```
ubuntuhost@workstation:~/CPE232Nagios$ ssh servercent
Last login: Mon Oct 24 20:58:18 2022 from 192.168.56.102
Ansible Managed node by Bello
[ubuntuhost@localhost ~]$ nagios

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

Website: https://www.nagios.org
Usage: nagios [options] <main_config_file>

Options:

  -v, --verify-config          Verify all configuration data (-v -v for more info)
  -s, --test-scheduling        Shows projected/recommended check scheduling and other
                               diagnostic info based on the current configuration files.
  -T, --enable-timing-point    Enable timed commentary on initialization
  -x, --dont-verify-paths      Deprecated (Don't check for circular object paths)
  -p, --precache-objects       Precache object configuration
  -u, --use-precached-objects  Use precached object config file
  -d, --daemon                 Starts Nagios in daemon mode, instead of as a foreground process
  -W, --worker /path/to/socket Act as a worker for an already running daemon

Visit the Nagios website at https://www.nagios.org/ for bug fixes, new
releases, online documentation, FAQs, information on subscribing to
the mailing lists, and commercial support options for Nagios.

[ubuntuhost@localhost ~]$
```

We used ssh to access both remote servers and as we can see both nagios are installed in the servers.



```
Connection to servercent closed.
ubuntuhost@workstation:~/CPE232Nagios$ git add -A
ubuntuhost@workstation:~/CPE232Nagios$ git commit -m "nagios"
[main (root-commit) ba2ff61] nagios
 4 files changed, 46 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 inventory
 create mode 100644 nagios.yml
 create mode 100644 roles/nagios/tasks/main.yml
ubuntuhost@workstation:~/CPE232Nagios$ git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (9/9), 858 bytes | 429.00 KiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qictbello/CPE232Nagios.git
 * [new branch]      main -> main
ubuntuhost@workstation:~/CPE232Nagios$ 
```

After completing the task don’t forget to push commit the changes into our repository.

**Reflections:**

Answer the following:

1. What are the benefits of having an availability monitoring tool?
   Monitoring networks using Ansible is efficient. We can implement many commands and fixes into our servers. We can check uptimes and modify and control our nodes by using ansible. We can also implement the use of GitHub so that in enterprise we can work as a team and implement changes in an efficient way.

**Conclusions:**

In conclusion, we made a playbook that installs nagios on both servers, Ubuntu and Centos, with the use of ansible, roles, and GitHub. We created this as a monitoring tool that will help us work on larger networks, for example. If a particular workload is active or reachable from our end, we utilize availability monitoring as a form of monitoring tool. Downtime on a website can result in loss of sales, harm to one's reputation, and major distress. By verifying the uptime of infrastructure elements like servers and apps and alerting the webmaster of issues before they have an impact on business, availability monitoring prevents undesirable scenarios.