Name: Hermitaño, Johnny C.	Date Performed: 10/20/2022
Course/Section: CPE31S23	Date Submitted: 10/20/2022
Instructor: Engr. Jonathan Taylar	Semester and SY: 1st sem sy 2022-2023

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)

Step 1. Enter the command **ssh-keygen** to generate an rsa key.

```
jhermitano@Workstation:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/jhermitano/.ssh/id_rsa):
/home/jhermitano/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jhermitano/.ssh/id_rsa
Your public key has been saved in /home/jhermitano/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:YRwMmMM8DwRolNqmDuHPOVg2qnJxN6SrAeZNw36qHKw jhermitano@Workstation
The key's randomart image is:
+---[RSA 3072]----+
|.oo=.o.o.        |
|.o  O  ...       |
|o.   =  +        |
|o +   o. .       |
|o= + o  S        |
|*o=++ o          |
|o+O=o+ .         |
|++o*+            |
|E+oo.            |
+----[SHA256]-----+
```

Step 2. Connect your control node to your manage node through ssh by entering the code **ssh-copy-id server@ip address** for Ubuntu and **ssh-copy-id -i ~/.ssh/id_rsa server@ip address** for CentOS.

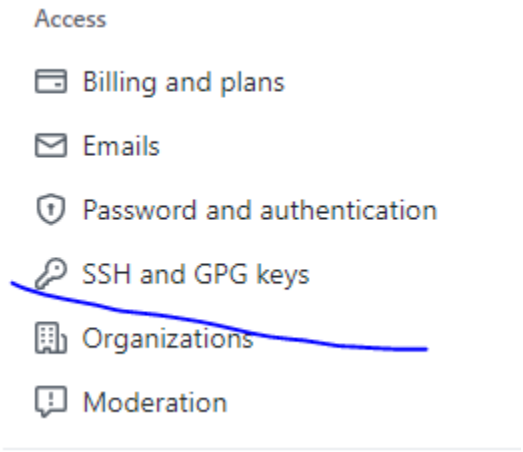
Ubuntu

```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ ssh-copy-id 192.168.56.105
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
 out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
jhermitano@192.168.56.105's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh '192.168.56.105'"
and check to make sure that only the key(s) you wanted were added.
```

CentOS

```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ ssh-copy-id -i ~/.ssh/id_rsa jhermi
tano@192.168.56.115
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jhermitano
/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
 out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist
 on the remote system.
                (if you think this is a mistake, you may want to use -f option)
```

Step 3. Connect your control node to your github account by adding your ssh key onto your github. To connect go to settings and click the **SSH and GPG keys.**



Just click the **New SSH keys** and then you may copy paste your rsa key from your control node. To see the rsa keys, enter **cat id_rsa.pub** and you will see your rsa keys.

```
jhermitano@Workstation:~/.ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCSZcs0YK385QI9dLl66GRWac+J1/Uc3D2tt0c4gH9
11mCna6vcqmLKfUSvyZKkEI0KI8rYvzoZSG6cvsSt9cQ3YK8F9PTUid1o/EenT7YvU7ehA+1R8e5aSb
bWu2zgvqaHIkvnqVoxHruehArk5ddMY4WF9toAHytZOANNOCVUuI33u8omdCFRFvHnxiq+UjccZKeGA
gbQ8GvW+5zC53pv9UDKOF/KUEHVX1wvt8nQRP1LmWmBvodnITs+EL7iwmtvKLD1KRdKtTYo+FfVrEjE
s5a8GEMaqeFPl6kMcAbwUdoliebIw+ZLBMRkmEwpywioG4kc74RT7Jh2NB3psWHF9fkJyItWPN4ZaZL
09n6wf5BT3cBzPI/fUgSGfhNc06le24I2cXaB3EFIu7pbiuTolDtfeEbov3m2FO3GFkaItutTpbyvgH
htQOGQReYG45obdBx46wREQVS+npF4kirHZYtkKystZf+afdTX8LcMBKOhTJmfMvJYulSY/6CQ7N0=
jhermitano@Workstation
```

Step 4. Clone your github repository by entering **git clone “your github repository link”**.

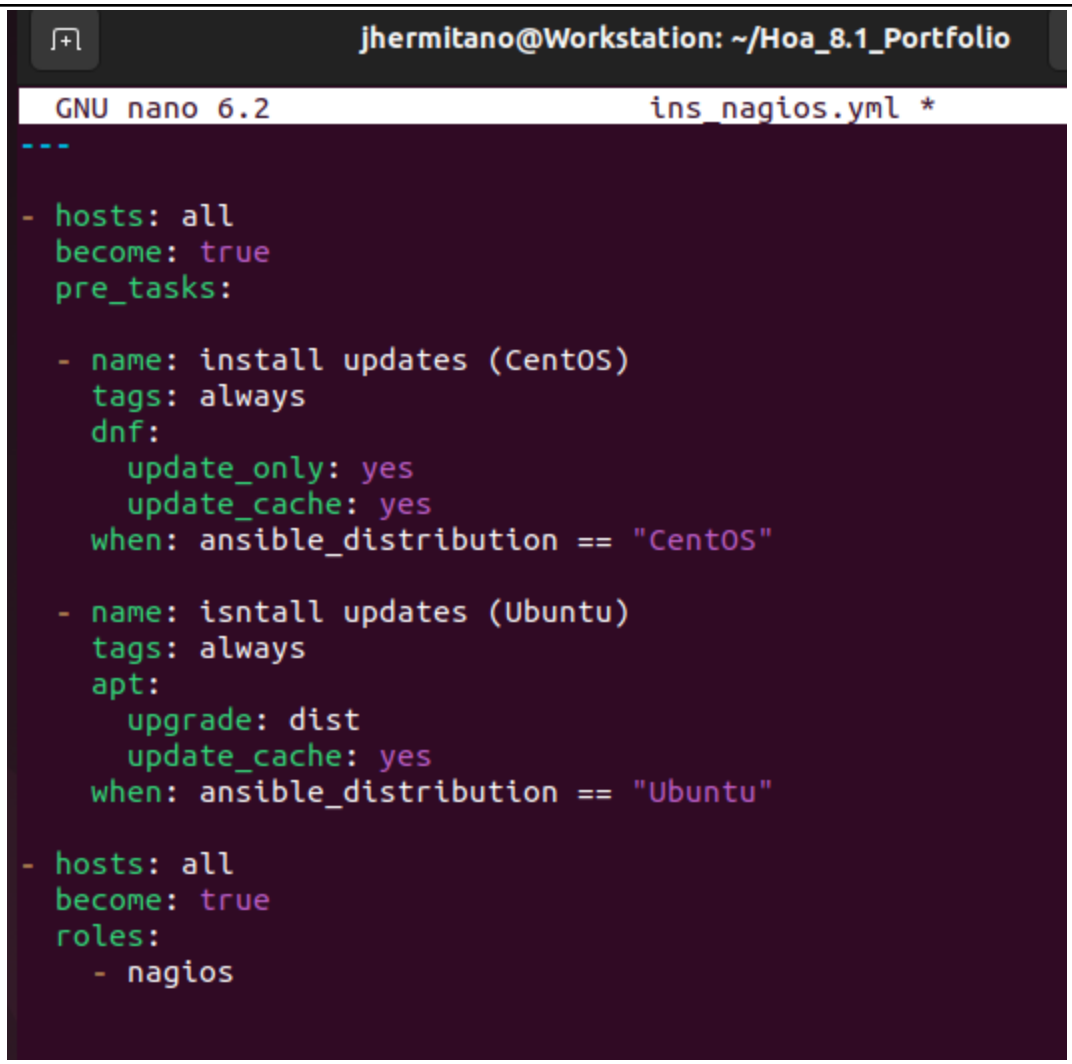
```
jhermitano@Workstation:~$ git clone git@github.com:jchermitano/Hoa_8.1_Portfoli
o.git
Cloning into 'Hoa_8.1_Portfolio'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

Step 5. Inside your repository, create your nano inventory and ansible.cfg for your playbook.

```
  GNU nano 6.2                                 inventory
[all]
192.168.56.105
192.168.56.115

[ubuntu]
192.168.56.105

[centos]
192.168.56.115
```

```
  GNU nano 6.2                                 ansible.cfg
[defaults]

inventory = inventory
Host_key_checking = False

depracation_warnings = False

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

Step 6. Create your playbook.



Step 7. Create a new directory named roles and inside it is another directory of your choice to put your main.yml. Then create your main.yml playbook.

```
jhermitano@Workstation: ~/Hoa_8.1_Portfolio/roles/nagios..
 GNU nano 6.2                                          main.yml
- name: nagios installation (Ubuntu)
  apt:
    name:
      - nagios4
    state: latest
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: nagios installation (CentOS)
  dnf:
    name:
      - nagios
    state: latest
    update_cache: yes
  when: ansible_distribution == "CentOS"
```

Your “tree” should look like this.

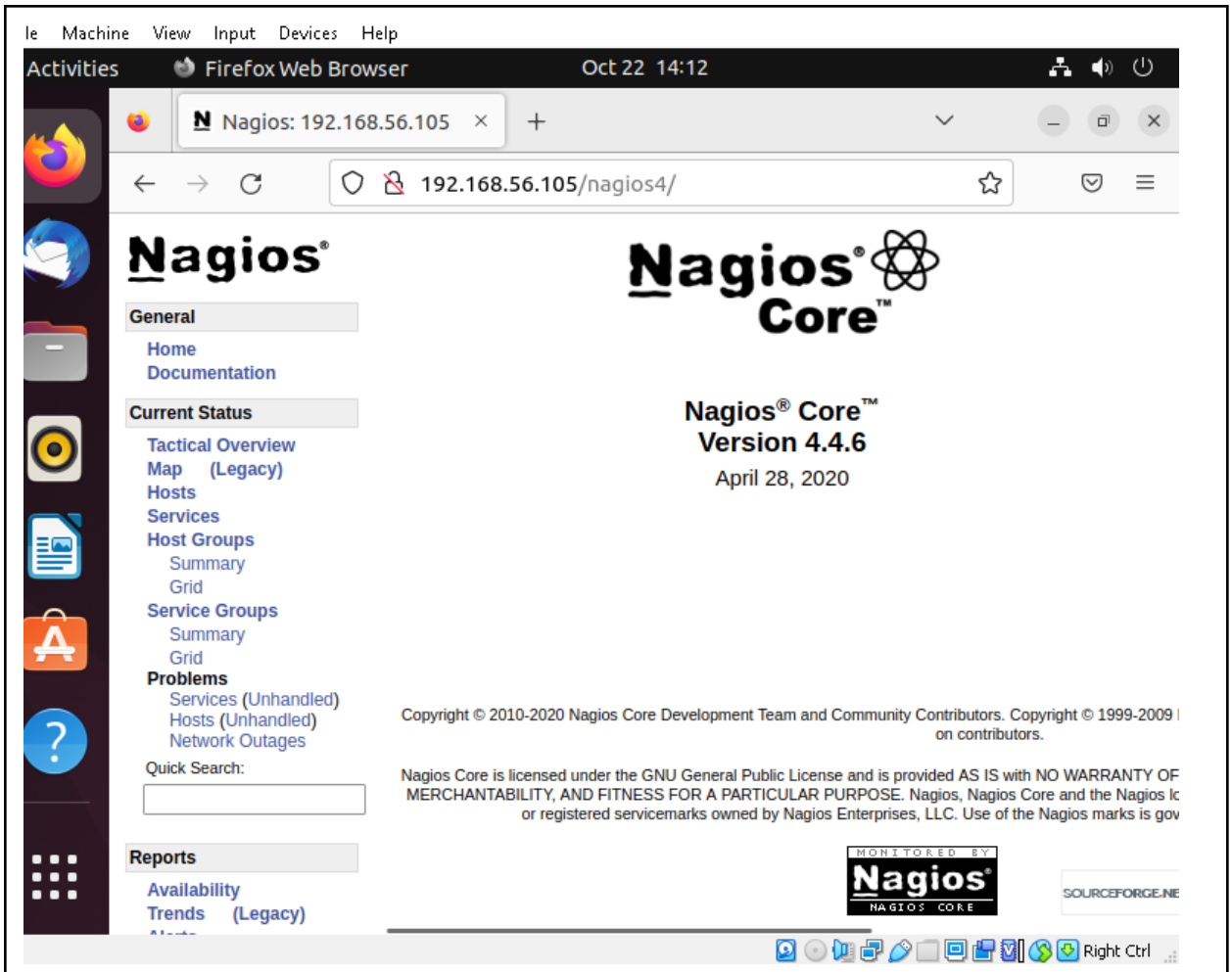
```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ tree
.
├── ansible.cfg
├── ins_nagios.yml
├── inventory
├── README.md
└── roles
    └── nagios
        └── tasks
            └── main.yml

3 directories, 5 files
```

After setting up your playbook, run it now with “ansible-playbook –ask-become-pass ‘playbook’” command.

```
jhermitano@Workstation:~/Hoa_8.1_Portfolio$ ansible-playbook --ask-become-pass ins_nagios.yml
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.105]
ok: [192.168.56.115]

TASK [install updates (CentOS)] ************************************************
skipping: [192.168.56.105]
ok: [192.168.56.115]

TASK [install updates (Ubuntu)] ************************************************
skipping: [192.168.56.115]
ok: [192.168.56.105]

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.56.105]
ok: [192.168.56.115]

TASK [nagios : nagios installation (Ubuntu)] ***********************************
skipping: [192.168.56.115]
ok: [192.168.56.105]

TASK [nagios : nagios installation (CentOS)] ***********************************
skipping: [192.168.56.105]
changed: [192.168.56.115]

PLAY RECAP *********************************************************************
192.168.56.105             : ok=4    changed=0    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
192.168.56.115             : ok=4    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0

jhermitano@Workstation:~/Hoa_8.1_Portfolio$
```

Step 8. Check your manage nodes if you successfully installed nagios by entering ipaddress/nagios4 (for ubuntu), ipaddress/nagios (for centos) into their browser.

Ubuntu:



CentOS: you can use other ways to check like entering command like “nagios -version”.

```
[jhermitano@localhost ~]$ nagios --version

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
This program is free software; you can redistribute it and/or modify
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.

[jhermitano@localhost ~]$
```

Reflections:

Answer the following:

1. What are the benefits of having an availability monitoring tool?
   In addition to providing information on trends, monitoring systems should be able to assist with capacity management by examining both network and device consumption. The ability to pinpoint locations where network bandwidth is being throttled makes this crucial to the operation.

Conclusions:

Through this project, I was able to build and develop a playbook that installs monitoring tools into CentOS and Ubuntu using ansible. Additionally, I have a better understanding of how Ansible's playbook and roles operate.

Github repository: https://github.com/jchermitano/Hoa_8.1_Portfolio.git