Name: Kevin Sumaya	Date Performed: October 9 2023
Course/Section: BSCPE 31S6	Date Submitted: October 12, 2023
Instructor: Dr. Jonathan Taylar	Semester and SY: 1st sem 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)

```
  GNU nano 2.9.3                    install_nagios.yml

# This is the playbook for installing Nagios on remote servers

---

- hosts: all
  become: true
  pre_tasks:

    - name: dnf and epel installation
      yum:
        name:
          - epel-release
          - dnf
      when: ansible_distribution == "CentOS"
```

- This code is an ansible playbook that installs all dnf and epel release packages on all hosts. This task uses the yum module to install the specified packages on CentOS hosts only, as specified by the when condition.

```
    - name: dpkg in ubuntu
      shell: |
        dpkg --configure -a
      when: ansible_distribution == "Ubuntu"

    - name: install updates (CentOS)
      dnf:
        update_cache: yes
        update_only: yes
      when: ansible_distribution == "CentOS"

    - name: install updates (Ubuntu)
      apt:
        upgrade: dist
        update_cache: yes
      when: ansible_distribution == "Ubuntu"
```

- The first code is an ansible playbook that runs the dpkg –configure -a command to ubuntu hosts. This command is used to configure any unpacked and unconfigured packages on the system.

```
- hosts: ubuntu_nagios
  become: true
  roles:
    - ubuntu_nagios

- hosts: centos_nagios
  become: true
  roles:
    - centos_nagios
```

- This is an ansible playbook that install and configure Nagios monitoring software on Ubuntu and CentOS hosts. The true command statement help in executing the tasks with elevated privileges.

CENTOS_NAGIOS

```
 GNU nano 2.9.3                          main.yml

- name: Installing nagios dependecies and libraries
  tags: dependecies, 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
```

```
      - net-snmp
      - net-snmp-utils
      - python2-pip
    state: latest

- name: Install passlib python package
  pip:
    name: passlib

- name: Creating a directory (where the downloaded files will be stored)
  file:
    path: ~/nagios
```

- This is an ansible playbook that installs the dependencies and libraries required for nagios monitoring software. the yum module helps to install the following packages specified in the inventory files like gcc , glibc, glibc-common etc.

- This ansible playbook install the passlib python package and creates a directory name nagios in the home directory of the current user. the pip module install the passlib package to all hosts specified.

```
    state: directory

- name: Downloading and extracting Nagios
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Compiling, installing, and adding users and groups in nagios
```

- this is the path where we get the download link to executes what file to download by the ubuntu terminal.

```
 GNU nano 2.9.3                          main.yml

    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
- name: Downloading and extracting Nagios plugins
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Compiling and installing plugins
  shell: |
    cd ~/nagios/nagios-plugins*
    ./tools/setup
```

- This block of codes is focused on downloading Nagios plugins and also compiling and installing the files in https://github.com/nagios -plugins.

```
    ./configure
    make
    make install
- name: Add a user to a password file and ensure permissions are set
  community.general.htpasswd:
    path: /usr/local/nagios/etc/htpasswd.users
    name: admin
    password: admin

- name: Making sure that nagios is started and enabled
  service:
    name: nagios
```

- This is where we put a password and name of the project for us to know what to input to sign in.

```
    state: restarted
    enabled: true

- name: Making sure that httpd is started and enabled
  service:
    name: httpd
    state: restarted
    enabled: true
```

- This block of codes helps to update the user if httpd is already started and enable its service.

UBUNTU_NAGIOS

```
 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
```

- This is an ansible playbook that installs the dependencies and libraries required for ubuntu monitoring software. the apt module helps to install the following packages specified in the inventory files like autoconf, libc6, gcc, make, make, wget etc.

```
      - gettext
      - python3
      - python3-pip
    state: latest

- name: passlib package
  pip:
    name: passlib

- name: nagios directory PATH
  file:
    path: ~/nagios
```

- The ansible playbook that installs the passlib python package and creates a directory named nagios in the home directory of the current user.

```
    state: directory

- name: downloading nagios
  unarchive:
    src: https://github.com/NagiosEnterprises/nagioscore/archive/nagios-4.4.6.$
    dest: ~/nagios
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: downloading nagios plugins
```

- This block of code helps in downloading the Nagios through a source destination which finds the correct Nagios version.

```
  unarchive:
    src: https://github.com/nagios-plugins/nagios-plugins/archive/release-2.3.$
    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
```

- This basically just install and compile the chosen nagios plugins and to put its chosen destination.

```
  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
```

- This creates a username and password user login for us to sign in our account.

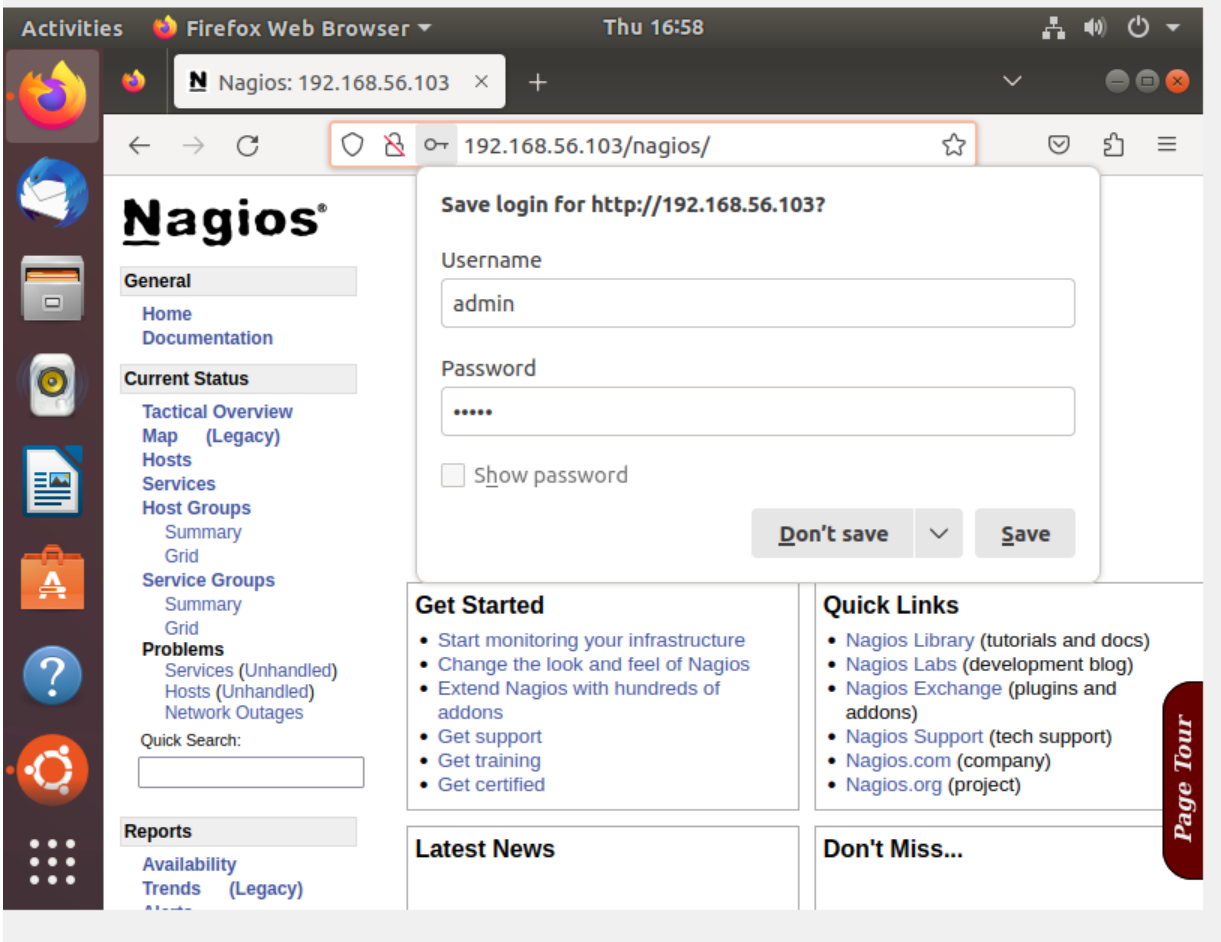
```
- name: Nagios Start/Enable Check
  service:
    name: nagios
    state: restarted
    enabled: true

- name: Apache/httpd Start/Enable check
  service:
    name: apache2
    state: restarted
    enabled: true
```

- This block of code focuses on restarting the Nagios service on all hosts specified in the inventory file. The service module is used to restart the nagios

and the enabled: true statement is used to ensure the service is enabled and starts automatically on time.

NAGIOS



- This is the Finished product

https://github.com/KevinS4160/Activity8

Reflections:

Answer the following:

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

- This helps organizations improve uptime, performance, security and cost effectiveness, all of which can contribute to increased productivity and customer satisfaction.

Conclusions:

- I learned many things in doing this Hands on activity like creating different playbooks to create an http site that only designated on a personal server. I also experienced different kinds of errors that I genuinely learned on how to debug. Since it detects the issue or errors that led to the program malfunctioning or shows which are the only functional components, monitoring is very helpful in this activity and simplifies the task. Having monitoring tools provides a number of benefits since you can determine which programs need to be fixed and if they are or are not.