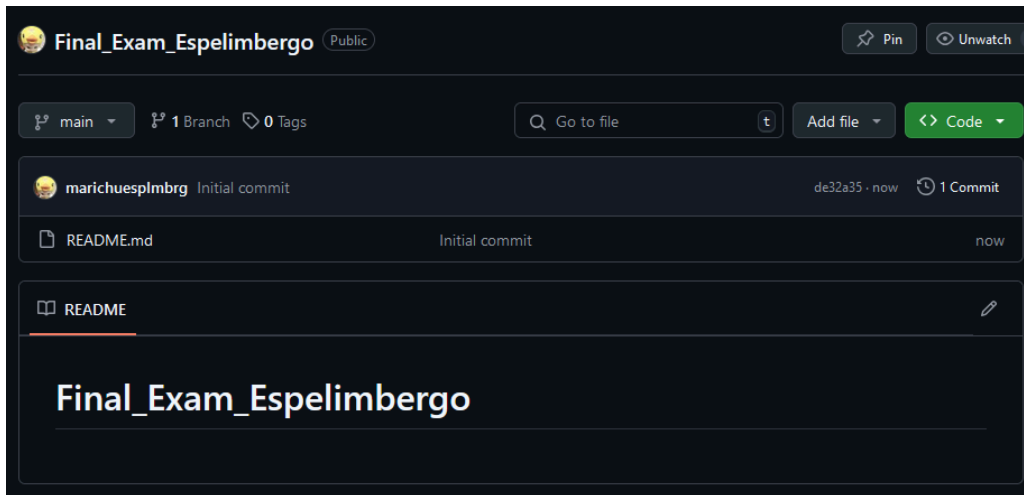


Name: Espelimbergo, Marichu .	Date: December 04, 2024
Course/Section: CPE212 – CPE31S2	Instructor: Engr. Robin Valenzuela
Tools Needed:	
1. VM with Ubuntu, CentOS and Ansible installed 2. Web browser	
Procedure:	
1. Create a repository and label it as "Final_Exam_Surname" 2. Clone your new repository in your VM 3. Create an Ansible playbook that does the following with an input of a config.yaml file and structure inventory file. 3.1 Install and configure one enterprise service that can be installed in Debian and Centos servers – <u>Apache</u> 3.2 Install and configure one monitoring tool that can be installed in Debian and Centos servers (if it is a stack there should be option of different host) – <u>Nagios</u> 3.3 Change Motd as "Ansible Managed by <username>" 4. Push and commit your files in GitHub 5. Make sure to show evidence of input (codes) process (codes successfully running) and output (evidence of installation) 6. For your final exam to be counted, please paste your repository link as an answer in this exam. <u>Note: Extra points if you will implement the said services via containerization.</u>	
Output:	



Cloned repository in Virtual Local Machine:

```
marichu@workstation:~/Final_Exam_Espelimbergo$ ls
ansible.cfg  config.yaml  inventory  README.md
marichu@workstation:~/Final_Exam_Espelimbergo$
```

```
GNU nano 2.9.3 ansible.cfg
[defaults]
inventory=/home/marichu/Final_Exam_Espelimbergo/inventory
remote_user=marichu
host_key_checking=True
```

- Input the path for the inventory that'll be used to install the tools in CentOS and Ubuntu servers.

```
GNU nano 2.9.3 inventory
[Ubuntu]
192.168.56.120

[CentOS]
192.168.56.121
```

- Input of the ip addresses for the ubuntu and CentOS servers.

CONFIGURATION:

```
config.yaml [Read-Only]
~/Final_Exam_Espelimbergo

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

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

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

  - name: install apache and php for Ubuntu Servers
    tags: apache,apache2,ubuntu
    apt:
      name:
        - apache2
        - libapache2-mod-php
      state: latest
      update_cache: yes
    when: ansible_distribution == "Ubuntu"
```

YAML ▼ Tab Width: 8 ▼ Ln 1, Col 1 ▼ INS

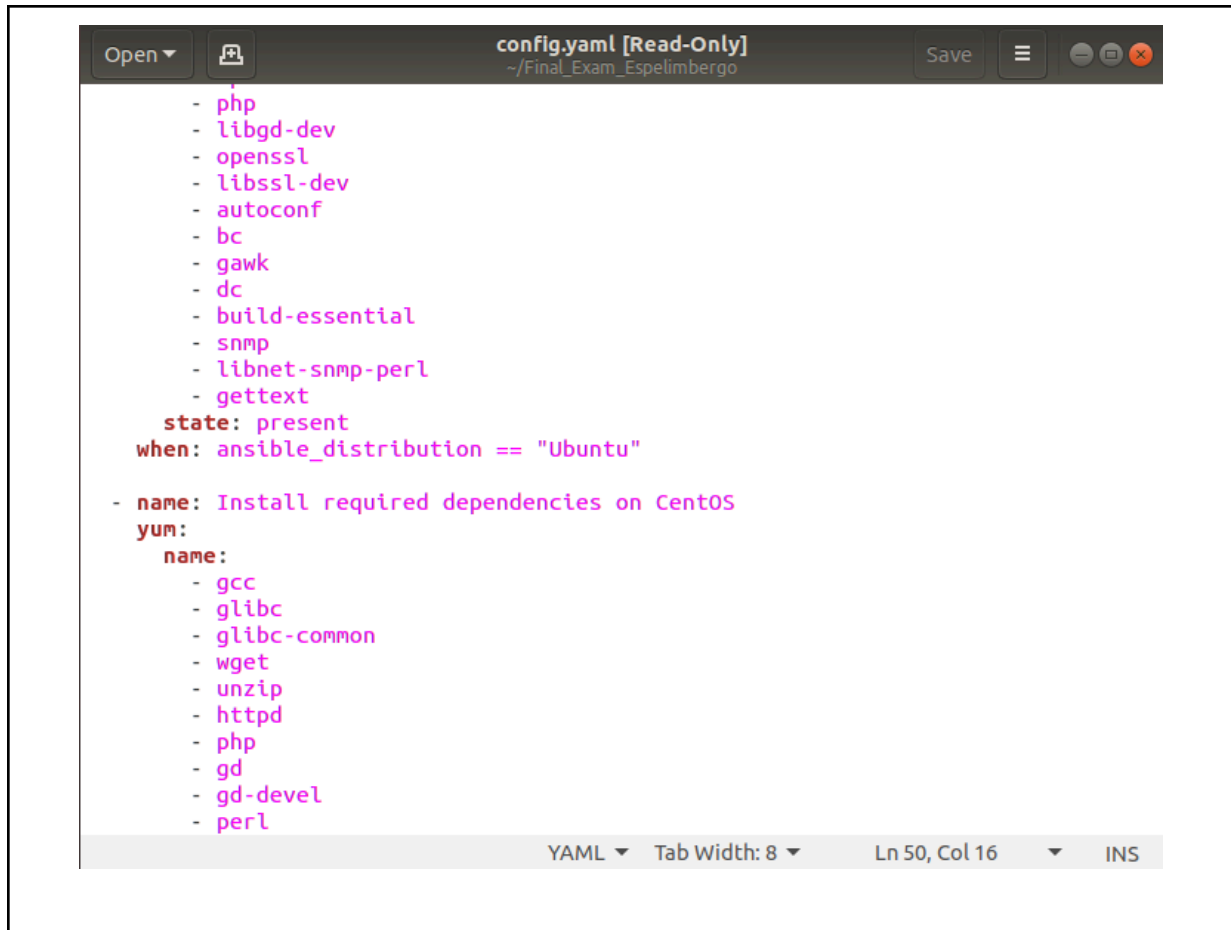
```
config.yaml [Read-Only]
~/Final_Exam_Espelimbergo

- name: install apache and php for CentOS servers
  tags: apache, centos, httpd
  dnf:
    name:
      - httpd
      - php
    state: latest
  when: ansible_distribution == "CentOS"

- name: start httpd (CentOS)
  tags: apache, centos, httpd
  service:
    name: httpd
    state: started
  when: ansible_distribution == "CentOS"

- name: Install required dependencies on Ubuntu
  apt:
    name:
      - gcc
      - libc6
      - make
      - wget
      - unzip
      - apache2
      - php
```

YAML ▾ Tab Width: 8 ▾ Ln 20, Col 15 ▾ INS






```
config.yaml [Read-Only]
~/Final_Exam_Espelimbergo

- php
- libgd-dev
- openssl
- libssl-dev
- autoconf
- bc
- gawk
- dc
- build-essential
- snmp
- libnet-snmp-perl
- gettext
  state: present
when: ansible_distribution == "Ubuntu"

- name: Install required dependencies on CentOS
  yum:
    name:
      - gcc
      - glibc
      - glibc-common
      - wget
      - unzip
      - httpd
      - php
      - gd
      - gd-devel
      - perl

YAML ▾ Tab Width: 8 ▾ Ln 50, Col 16 ▾ INS
```

```
Open ▾  config.yaml [Read-Only] Save   
~/Final_Exam_Espelimbergo

- name: Install Nagios dependencies
  package:
    - gcc-devel
    - perl
    - postfix
    - openssl
    - openssl-devel
    - make
    - autoconf
  state: present
  when: ansible_distribution == "CentOS"


- name: Download Nagios Core source code
  get_url:
    url: "https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.5.6.tar.gz"
    dest: /tmp/nagios-4.5.6.tar.gz

- name: Extract Nagios source code
  unarchive:
    src: /tmp/nagios-4.5.6.tar.gz
    dest: /tmp
    remote_src: yes




- name: Download Nagios Plugins
  get_url:
    url: "https://nagios-plugins.org/download/nagios-plugins-2.4.11.tar.gz"
    dest: /tmp/nagios-plugins-2.4.11.tar.gz

- name: Extract Nagios Plugins
  unarchive:
```

YAML ▾ Tab Width: 8 ▾ Ln 76, Col 15 ▾ INS

Open ▾

config.yaml [Read-Only]
~/Final_Exam_Espelimbergo

Save

```
- name: Extract Nagios Plugins
  unarchive:
    src: /tmp/nagios-plugins-2.4.11.tar.gz
    dest: /tmp
    remote_src: yes

- name: Create Nagios group
  group:
    name: nagios

- name: Create Nagios user and group
  user:
    name: nagios
    group: nagios

- name: Create nagcmd group
  group:
    name: nagcmd

- name: Add nagios and apache/httpd users to nagcmd group
  user:
    name: "{{ item }}"
    groups: nagcmd
    append: yes
  loop:
    - nagios
    - "{{ 'www-data' if ansible_os_family == 'Debian' else 'apache' }}"
```

YAML ▾ Tab Width: 8 ▾ Ln 76, Col 15 ▾ INS

```
config.yaml [Read-Only]
~/Final_Exam_Espelimbergo

loop:
  - nagios
  - "{{ 'www-data' if ansible_os_family == 'Debian' else 'apache' }}"

- name: Compile and install Nagios Core
  shell: |
    cd /tmp/nagios-4.5.6
    ./configure --with-command-group=nagcmd
    make all
    make install
    make install-init
    make install-commandmode
    make install-config
    make install-webconf
  args:
    creates: /usr/local/nagios/bin/nagios

- name: Install Nagios Plugins
  shell: |
    cd /tmp/nagios-plugins-2.4.11
    ./configure --with-nagios-user=nagios --with-nagios-group=nagios
    make
    make install
  args:
    creates: /usr/local/nagios/libexec/check_http

- name: Set Nagios admin password
  command: httpasswd -b -c /usr/local/nagios/etc/httpasswd.users marichu
```

YAML ▼ Tab Width: 8 ▼ Ln 76, Col 15 ▼ INS


```
config.yaml [Read-Only]
~/Final_Exam_Espelimbergo
Open Save

creates: /usr/local/nagios/libexec/check_http

- name: Set Nagios admin password
  command: htpasswd -b -c /usr/local/nagios/etc/htpasswd.users marichu
"monstaxbts"

- name: Enable and start Apache/Httpd service on Ubuntu
  service:
    name: apache2
    enabled: yes
    state: started
  when: ansible_distribution == "Ubuntu"

- name: Enable and start Apache/Httpd service on CentOS
  service:
    name: httpd
    enabled: yes
    state: started
  when: ansible_distribution == "CentOS"

- name: Enable and start Nagios service
  service:
    name: nagios
    enabled: yes
    state: started

- name: Enable external command execution in Nagios
  lineinfile:
    path: /usr/local/nagios/etc/nagios.cfg
    line: enable_external_commands=1
    create: yes
    mode: 0644
```

YAML Tab Width: 8 Ln 76, Col 15 INS

```
Open ▾  config.yaml [Read-Only]  Save  ⋮  ⌵  ✖
~/Final_Exam_Espelimbergo

- name: Enable and start Nagios service
  service:
    name: nagios
    enabled: yes
    state: started

- name: Enable external command execution in Nagios
  lineinfile:
    path: /usr/local/nagios/etc/nagios.cfg
    regexp: '^#?check_external_commands='
    line: 'check_external_commands=1'

- name: Restart Nagios service to apply changes
  service:
    name: nagios
    state: restarted

- name: Restart Apache/Httpd to apply changes on Ubuntu
  service:
    name: apache2
    state: restarted
  when: ansible_distribution == "Ubuntu"

- name: Restart Apache/Httpd to apply changes on CentOS
  service:
    name: httpd
    state: restarted
  when: ansible_distribution == "CentOS"
```

- name: Set MOTD script
copy:
content: "Ansible managed by marichu"
dest: /etc/motd
owner: root
group: root
mode: 0644

- Configuration for the installation of the enterprise service and the monitoring tool.

RESULTS:

```
marichu@workstation:~/Final_Exam_Espellmbergo$ ansible-playbook --ask-become-pass config.yaml
BECOME password:
```

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

```
TASK [Gathering Facts] *****
*
```

```
[DEPRECATION WARNING]: Distribution centos 9 on host 192.168.56.121 should use /usr/libexec/platform-python, but is using /usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using the discovered platform python for this host. See https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more information. This feature will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
```

```
ok: [192.168.56.121]
```

```
[DEPRECATION WARNING]: Distribution Ubuntu 18.04 on host 192.168.56.120 should use /usr/bin/python3, but is using /usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using the discovered platform python for this host. See https://docs.ansible.com/ansible/2.9/reference\_appendices/interpreter\_discovery.html for more information. This feature will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
```

```
ok: [192.168.56.120]
```

```
TASK [install updates (CentOS)] *****
*
```

```
skipping: [192.168.56.120]
```

```
ok: [192.168.56.121]
```

```
TASK [install updates (Ubuntu)] *****
*
```

```
skipping: [192.168.56.121]
```

```
ok: [192.168.56.120]
```

```
TASK [install apache and php for Ubuntu Servers] *****
*
```

```
skipping: [192.168.56.121]
```

```
ok: [192.168.56.120]
```

```
TASK [install apache and php for CentOS servers] *****
*
```

```
skipping: [192.168.56.120]
```

```
ok: [192.168.56.121]
```

```
TASK [start httpd (CentOS)] *****
*
```

```
skipping: [192.168.56.120]
```

```
ok: [192.168.56.121]
```

```
TASK [start httpd (CentOS)] *****
*
skipping: [192.168.56.120]
ok: [192.168.56.121]

TASK [Install required dependencies on Ubuntu] *****
*
skipping: [192.168.56.121]
ok: [192.168.56.120]

TASK [Install required dependencies on CentOS] *****
*
skipping: [192.168.56.120]
ok: [192.168.56.121]

TASK [Download Nagios Core source code] *****
*
ok: [192.168.56.121]
changed: [192.168.56.120]

TASK [Extract Nagios source code] *****
*
changed: [192.168.56.120]
ok: [192.168.56.121]

TASK [Download Nagios Plugins] *****
*

TASK [Download Nagios Plugins] *****
*
ok: [192.168.56.121]
changed: [192.168.56.120]

TASK [Extract Nagios Plugins] *****
*
changed: [192.168.56.120]
ok: [192.168.56.121]

TASK [Create Nagios group] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Create Nagios user and group] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Create nagcmd group] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Add nagios and apache/httpd users to nagcmd group] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]
```

```
TASK [Add nagios and apache/httpd users to nagcmd group] *****
*
ok: [192.168.56.120] => (item=nagios)
ok: [192.168.56.120] => (item=www-data)
ok: [192.168.56.121] => (item=nagios)
ok: [192.168.56.121] => (item=apache)

TASK [Compile and install Nagios Core] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Install Nagios Plugins] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Set Nagios admin password] *****
*
changed: [192.168.56.120]
changed: [192.168.56.121]

TASK [Enable and start Apache/Httpd service on Ubuntu] *****
*
skipping: [192.168.56.121]
ok: [192.168.56.120]
```

```
TASK [Enable and start Apache/Httpd service on Ubuntu] *****
*
skipping: [192.168.56.121]
ok: [192.168.56.120]

TASK [Enable and start Apache/Httpd service on CentOS] *****
*
skipping: [192.168.56.120]
ok: [192.168.56.121]

TASK [Enable and start Nagios service] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Enable external command execution in Nagios] *****
*
ok: [192.168.56.120]
ok: [192.168.56.121]

TASK [Restart Nagios service to apply changes] *****
*
changed: [192.168.56.120]
changed: [192.168.56.121]

TASK [Restart Apache/Httpd to apply changes on Ubuntu] *****
*
```

```

TASK [Restart Apache/Httpd to apply changes on Ubuntu] *****
*
skipping: [192.168.56.121]
changed: [192.168.56.120]

TASK [Restart Apache/Httpd to apply changes on CentOS] *****
*
skipping: [192.168.56.120]
changed: [192.168.56.121]

PLAY RECAP *****
*
192.168.56.120      : ok=20    changed=7    unreachable=0    failed=0
skipped=6      rescued=0    ignored=0
192.168.56.121      : ok=21    changed=3    unreachable=0    failed=0
skipped=5      rescued=0    ignored=0

marichu@workstation:~/Final_Exam_Espelimbergo$

```

- Result of the playbook after I run it in my Local Machine.

UBUNTU:

```

marichu@server1:~$ sudo systemctl status apache2.service
[sudo] password for marichu:
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since Wed 2024-12-04 09:10:06 +01; 1min ago
     Process: 8081 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS)
     Process: 8090 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 8094 (apache2)
      Tasks: 6 (limit: 2260)
   CGroup: /system.slice/apache2.service
           └─8094 /usr/sbin/apache2 -k start
             └─8099 /usr/sbin/apache2 -k start
               └─8100 /usr/sbin/apache2 -k start
                 └─8101 /usr/sbin/apache2 -k start
                   └─8102 /usr/sbin/apache2 -k start
                     └─8103 /usr/sbin/apache2 -k start

Dec 04 09:10:06 server1 systemd[1]: Starting The Apache HTTP Server:
Dec 04 09:10:06 server1 apachectl[8090]: AH00558: apache2: Syntax error on line 1 of /etc/apache2/httpd.conf: Invalid command 'LoadModule',
Dec 04 09:10:06 server1 systemd[1]: Started The Apache HTTP Server:
lines 1-20/20 (END)

```

```

marichu@server1:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.6
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Wed 2024-12-04 09:10:03 +01; 1min ago
     Docs: https://www.nagios.org/documentation
   Process: 8040 ExecStopPost=/bin/rm -f /usr/local/nagios/var/*
   Process: 8038 ExecStop=/bin/kill -s TERM ${MAINPID} (code=0)
   Process: 8042 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
   Process: 8041 ExecStartPre=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
 Main PID: 8043 (nagios)
    Tasks: 8 (limit: 2260)
   CGroup: /system.slice/nagios.service
           └─8043 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
           └─8044 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8045 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8046 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8047 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8048 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8049 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/etc/nagios.cfg
           └─8055 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Dec 04 09:10:03 server1 nagios[8043]: qh: echo service query
Dec 04 09:10:03 server1 nagios[8043]: qh: help for the query
Dec 04 09:10:03 server1 nagios[8043]: wproc: Successfully received
lines 1-23...skipping...

```

- This shows that apache and nagios are running and are active in the Server 1 Ubuntu.

CentOS:

```

[marichu@localhost ~]$ sudo systemctl status httpd
[sudo] password for marichu:
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/httpd.service.d
            └─php-fpm.conf
   Active: active (running) since Wed 2024-12-04 09:10:09 PST; 18min ago
     Docs: man:httpd.service(8)
 Main PID: 13167 (httpd)
   Status: "Total requests: 4; Idle/Busy workers 100/0;Requests/sec: 0.0036;..."
    Tasks: 177 (limit: 10945)
   Memory: 34.5M
      CPU: 2.416s
   CGroup: /system.slice/httpd.service
           └─13167 /usr/sbin/httpd -DFOREGROUND
           └─13169 /usr/sbin/httpd -DFOREGROUND
           └─13170 /usr/sbin/httpd -DFOREGROUND
           └─13171 /usr/sbin/httpd -DFOREGROUND
           └─13172 /usr/sbin/httpd -DFOREGROUND

Dec 04 09:10:08 localhost.localdomain systemd[1]: Starting The Apache HTTP Server:
Dec 04 09:10:09 localhost.localdomain httpd[13167]: AH00558: httpd: Could not r
Dec 04 09:10:09 localhost.localdomain httpd[13167]: Server configured, listenin
Dec 04 09:10:09 localhost.localdomain systemd[1]: Started The Apache HTTP Serve
lines 1-22/22 (END)

```

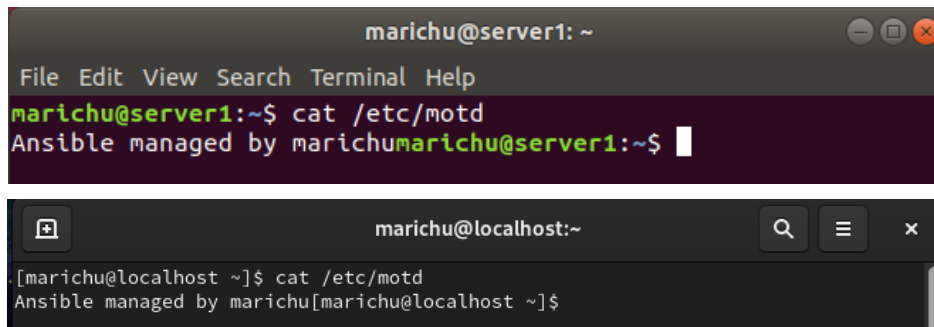
```
[marichu@localhost ~]$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.5.6
   Loaded: loaded (/usr/lib/systemd/system/nagios.service; enabled; preset: disabled)
   Active: active (running) since Wed 2024-12-04 09:10:03 PST; 20min ago
     Docs: https://www.nagios.org/documentation
   Process: 13014 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
   Process: 13015 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
   Main PID: 13016 (nagios)
    Tasks: 8 (limit: 10945)
   Memory: 10.8M
      CPU: 747ms
   CGroup: /system.slice/nagios.service
           └─13016 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
             └─13017 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
               └─13018 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
                 └─13019 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
                   └─13020 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
                     └─13021 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
                       └─13022 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw
                         └─13044 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Dec 04 09:10:03 localhost.localdomain nagios[13016]: wproc: Registry request: name=>
Dec 04 09:10:03 localhost.localdomain nagios[13016]: Successfully launched command >
Dec 04 09:10:07 localhost.localdomain nagios[13016]: SERVICE ALERT: localhost;Swap >
Dec 04 09:11:07 localhost.localdomain nagios[13016]: SERVICE ALERT: localhost;Swap >
Dec 04 09:12:07 localhost.localdomain nagios[13016]: SERVICE NOTIFICATION: nagiosad>
Dec 04 09:12:07 localhost.localdomain nagios[13016]: SERVICE ALERT: localhost;Swap >
lines 1-26...skipping...
```

- This shows that apache and nagios are running and are active in the CentOS.

MOTD:

Change Motd as "Ansible Managed by <marichu>"



```
marichu@server1: ~
File Edit View Search Terminal Help
marichu@server1:~$ cat /etc/motd
Ansible managed by marichumarichu@server1:~$
```

```
marichu@localhost:~
[marichu@localhost ~]$ cat /etc/motd
Ansible managed by marichumarichu@localhost ~]$
```

REPOSITORY (ADD, COMMIT, AND PUS):

```
marichu@workstation:~/Final_Exam_Espelimbergo$ git add *
```



```
marichu@workstation:~/Final_Exam_Espelimbergo$ git push
Username for 'https://github.com': marichuesplmbrg
Password for 'https://marichuesplmbrg@github.com':
Counting objects: 12, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 2.47 KiB | 2.47 MiB/s, done.
Total 12 (delta 5), reused 0 (delta 0)
remote: Resolving deltas: 100% (5/5), done.
To https://github.com/marichuesplmbrg/Final_Exam_Espelimbergo.git
de32a35..1eb0936 main -> main
marichu@workstation:~/Final_Exam_Espelimbergo$
```

The screenshot shows the GitHub interface for a repository named 'Final_Exam_Espelimbergo'. The repository is public and has 1 branch (main) and 0 tags. The commit history shows 4 commits by user 'marichuesplmbrg'. The commit '1eb0936' from 4 minutes ago is highlighted, with the message 'Remove unnecessary config files'. Below the commit list, a table shows the files included in the commit:

File	Commit	Time
README.md	Initial commit	2 hours ago
ansible.cfg	Final_Exam	13 minutes ago
config.yaml	Final_Exam	13 minutes ago
config.yaml.save	Finals	10 minutes ago
inventory	Final_Exam	13 minutes ago

CONCLUSION:

For this Finals Skills Exam we are tasked to configure one enterprise service and one monitoring tool just like the past activities we had before. It is not that hard unlike the past skills exam because like I've said we have done this kind of activity just with setting up the MOTD script inside the config.yaml.