

### Tools Needed:

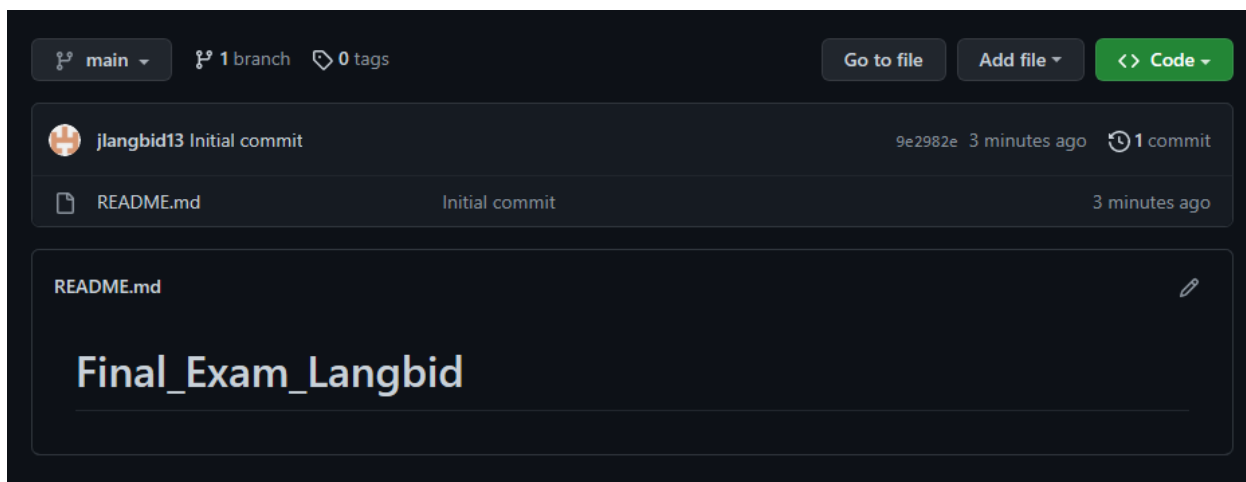
1. VM with Ubuntu, CentOS and Ansible installed
2. Web browser

### 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
  - 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)
- 4.4 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). Create a word document report for this final exam. For your final exam to be counted, please paste your repository link as an answer in your report. No point will be given if you forgot to paste your repo link.

Note: Extra points if you will implement the said services via containerization.

1. Create a repository and label it as "Final\_Exam\_Surname"



Jefferson Langbid

## 2. Clone your new repository in your VM

```
jefferson@LocalMachine: ~  
jefferson@LocalMachine:~$ git clone git@github.com:jangbid13/Final_Exam_Langbi  
d  
Cloning into 'Final_Exam_Langbid'...  
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.  
jefferson@LocalMachine:~$
```

```
jefferson@LocalMachine: ~/Final_Exam_Langbid  
GNU nano 6.2 ansible.cfg  
[defaults]  
  
inventory = inventory  
host_key_changing = False  
  
deprecation_warnings = False  
  
remote_user = jefferson  
private_key_file = ~/.ssh/
```

```
jefferson@LocalMachine: ~/Final_Exam_Langbid  
GNU nano 6.2 inventory  
[remote_servers]  
192.168.56.104  
192.168.56.106  
  
[centos]  
192.168.56.106  
  
[ubuntu]  
192.168.56.104
```

Jefferson Langbid

3. Create an Ansible playbook that does the following with an input of a config.yaml file and structure inventory file.

```
jefferson@LocalMachine: ~/Final_Exam_Langbid
GNU nano 6.2 config.yaml
---
- hosts: all
  become: true
  pre_tasks:
    - name: Install Update (Ubuntu)
      apt:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "Ubuntu"
    - name: Update repository index (CentOS)
      dnf:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "CentOS"
```

```
jefferson@LocalMachine:~/Final_Exam_Langbid/roles$ tree
.
├── enterprise
│   └── tasks
│       └── main.yml
└── monitor
    └── tasks
        └── main.yml

4 directories, 2 files
```

3.1 Install and configure one enterprise service that can be installed in Debian and Centos servers

```
jefferson@LocalMachine: ~/Final_Exam_Langbid/roles/ente...
GNU nano 6.2 main.yml
- name: install apache and php for Ubuntu
  apt:
    name:
      - apache2
      - libapache2-mod-php
    state: present
    update_cache: yes
    when: ansible_distribution == "Ubuntu"

- name: install apache and php for CentOS
  yum:
    name:
      - httpd
      - php
    state: present
    update_cache: yes
    when: ansible_distribution == "CentOS"
```

```
jefferson@LocalMachine: ~/Final_Exam_Langbid
PLAY [exam_server] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.104]
ok: [192.168.56.106]

TASK [enterprise : install apache and php for Ubuntu] *****
*
skipping: [192.168.56.106]
ok: [192.168.56.104]

TASK [enterprise : install apache and php for CentOS] *****
*
skipping: [192.168.56.104]
ok: [192.168.56.106]
```

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)

```
jefferson@LocalMachine: ~/Final_Exam_Langbid/roles/moni...
GNU nano 6.2 main.yml
- name: install prometheus in Ubuntu
  apt:
    name:
      - prometheus
    state: present
    update_cache: yes
  when: ansible_distribution == "Ubuntu"

- name: install prometheus requisites in CentOS
  tags: centos, snapd, epel-release
  dnf:
    name:
      - epel-release
      - snapd
    state: latest
  when: ansible_distribution == "CentOS"

- name: Enabling sockets for CentOS
  tags: snapd, centos
  command: systemctl enable --now snapd.socket
  when: ansible_distribution == "CentOS"

- name: Finising installation of Prometheus for CentOS
  tags: centos, prometheus
  command: snap install prometheus --classic

[ Read 26 lines ]

TASK [monitor : install prometheus in Ubuntu] *****
*
skipping: [192.168.56.106]
ok: [192.168.56.104]

TASK [monitor : install prometheus requisites in CentOS] *****
*
skipping: [192.168.56.104]
ok: [192.168.56.106]

TASK [monitor : Enabling sockets for CentOS] *****
*
skipping: [192.168.56.104]
changed: [192.168.56.106]

TASK [monitor : Finising installation of Prometheus for CentOS] *****
*
skipping: [192.168.56.104]
changed: [192.168.56.106]
```

#### 4.4 Change Motd as "Ansible Managed by <username>"

```
- name: Creating Banner Motd in Ubuntu
  debug:
    msg: "Ansible Managed node by Langbid"
  when: ansible_distribution == "Ubuntu"

- name: Creating Banner Motd in CentOS
  debug:
    msg: "Ansible Managed node by Langbid"
  when: ansible_distribution == "CentOS"
```

```
TASK [Gathering Facts] *****
*
ok: [192.168.56.104]
ok: [192.168.56.106]

TASK [Creating Banner Motd in Ubuntu] *****
*
skipping: [192.168.56.106]
ok: [192.168.56.104] => {
  "msg": "Ansible Managed node by Langbid"
}

TASK [Creating Banner Motd in CentOS] *****
*
skipping: [192.168.56.104]
ok: [192.168.56.106] => {
  "msg": "Ansible Managed node by Langbid"
}

PLAY RECAP *****
*
192.168.56.104      : ok=7    changed=0    unreachable=0    failed=0
skipped=6    rescued=0    ignored=0
192.168.56.106      : ok=9    changed=2    unreachable=0    failed=0
skipped=4    rescued=0    ignored=0
```

The code ran successfully and it installed the enterprise and monitoring service and tools.

Jefferson Langbid

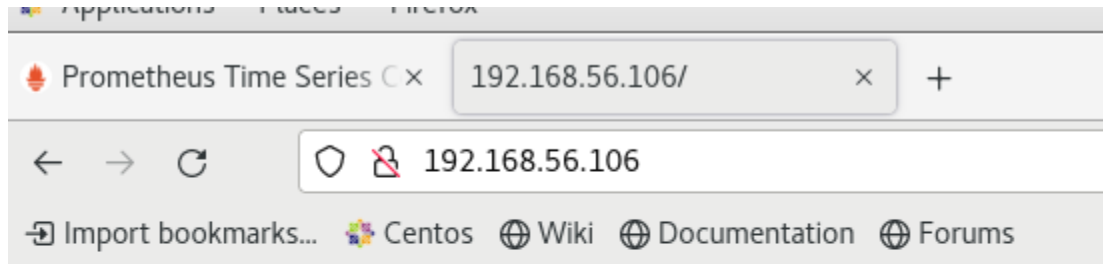
Proof that the enterprise service and monitoring tool is installed.

Ubuntu

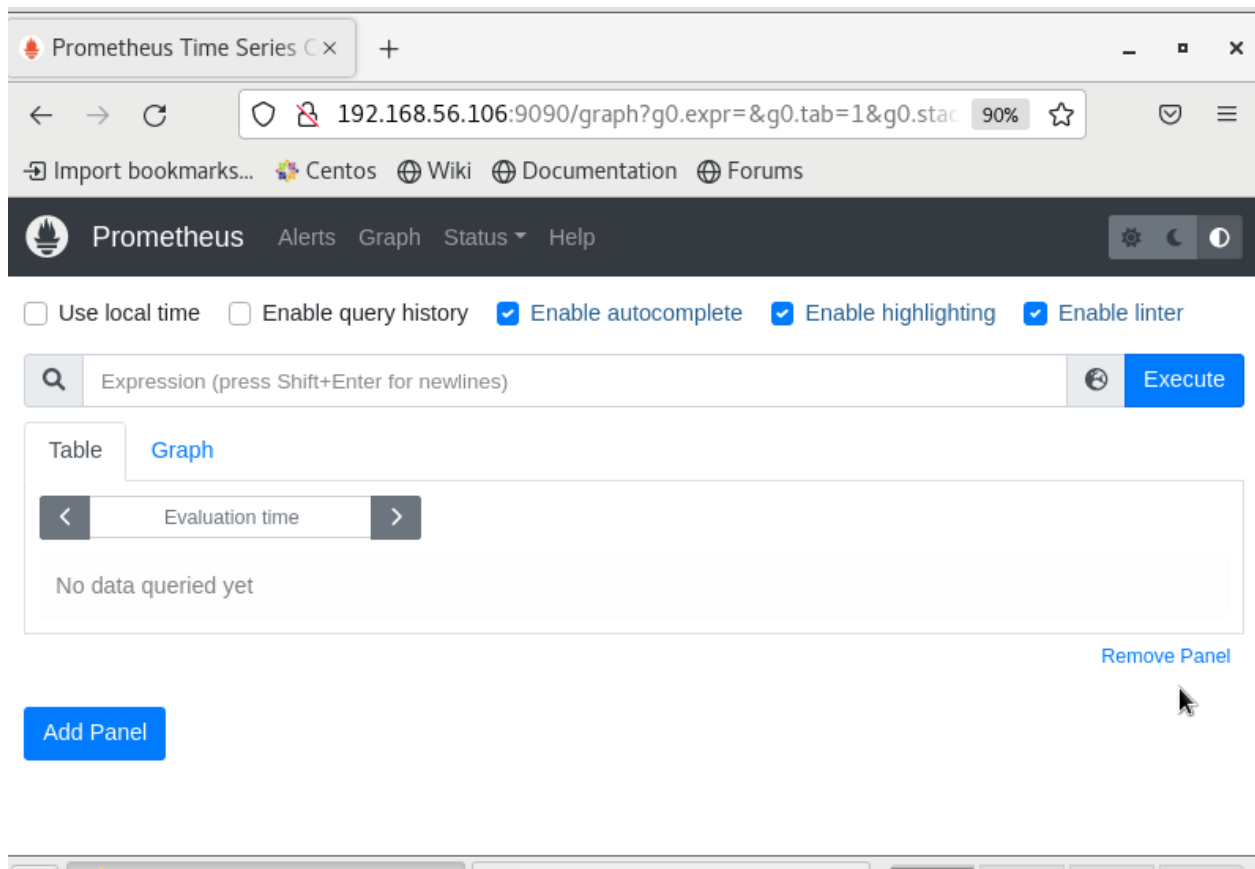
The screenshot displays two browser windows. The top window, titled 'Apache2 Ubuntu Default Page', shows the default Apache2 welcome page on the IP address 192.168.56.104. The page features the Ubuntu logo, the text 'Apache2 Default Page', and a red button that says 'It works!'. Below this, there is a 'Configuration Overview' section with text explaining the default configuration and a code block showing the path `/etc/apache2/` and the file `apache2.conf`.

The bottom window, titled 'Prometheus Time Series', shows the Prometheus web interface on the IP address 192.168.56.104:9090/classic/graph. The interface includes a navigation bar with 'Prometheus', 'Alerts', 'Graph', 'Status', and 'Help'. Below the navigation bar, there is a checkbox for 'Enable query history' and a text input field for the query expression. A blue 'Execute' button is next to the input field, and a dropdown menu shows '- insert metric at cursor -'. To the right of the input field is a blue link that says 'Remove Graph'. Below the input field, there are two tabs: 'Graph' and 'Console'. The 'Graph' tab is selected, and it shows a time range selector set to 'Moment' with left and right arrow buttons. Below the time range selector, there is a table with two columns: 'Element' and 'Value'. The table currently shows 'no data'. At the bottom left of the interface is a blue button that says 'Add Graph'.

## CentOS



## Ansible





```

Unknown operation: httpd.
[jefferson@localhost ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2022-12-15 10:39:21 PST; 10min ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Main PID: 26260 (httpd)
    Status: "Total requests: 4; Current requests/sec: 0; Current traffic: 0 B/sec"
     Tasks: 7
    CGroup: /system.slice/httpd.service
            └─26260 /usr/sbin/httpd -DFOREGROUND
              └─26272 /usr/sbin/httpd -DFOREGROUND
                └─26273 /usr/sbin/httpd -DFOREGROUND
                  └─26274 /usr/sbin/httpd -DFOREGROUND
                    └─26275 /usr/sbin/httpd -DFOREGROUND
                      └─26276 /usr/sbin/httpd -DFOREGROUND
                        └─27747 /usr/sbin/httpd -DFOREGROUND

Dec 15 10:39:20 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Dec 15 10:39:21 localhost.localdomain httpd[26260]: AH00558: httpd: Could not reliably determine the server's fully qualified domain name, because the 'ServerName' directive was not
Dec 15 10:39:21 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.

```

```
jefferson@Server2: ~$ sudo systemctl status apache2
```

```
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enable)  
   Active: active (running) since Thu 2022-12-15 10:01:20 PST; 1h 7min ago  
     Docs: https://httpd.apache.org/docs/2.4/  
    Main PID: 1180 (apache2)  
      Tasks: 47 (limit: 1075)  
    Memory: 8.2M  
       CPU: 3.428s  
    CGroup: /system.slice/apache2.service  
            └─ 1180 /usr/sbin/apache2 -k start  
              └─ 1339 "(wsgi:cinder-wsgi" -k start  
                ├── 1340 "(wsgi:cinder-wsgi" -k start  
                ├── 1341 "(wsgi:cinder-wsgi" -k start  
                ├── 1347 "(wsgi:cinder-wsgi" -k start  
                ├── 1348 "(wsgi:cinder-wsgi" -k start  
                ├── 1349 "(wsgi:keystone-pu" -k start  
                ├── 1350 "(wsgi:keystone-pu" -k start  
                ├── 1351 "(wsgi:keystone-pu" -k start  
                ├── 1352 "(wsgi:keystone-pu" -k start  
                ├── 1353 "(wsgi:keystone-pu" -k start  
                ├── 1354 /usr/sbin/apache2 -k start  
                ├── 1355 /usr/sbin/apache2 -k start  
                ├── 1356 /usr/sbin/apache2 -k start  
                ├── 1357 /usr/sbin/apache2 -k start  
                ├── 1358 /usr/sbin/apache2 -k start  
                └─ 13926 /usr/sbin/apache2 -k start
```

#### 4. Push and commit your files in GitHub

```
jefferson@LocalMachine:~/Final_Exam_Langbid$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   roles/enterprise/tasks/main.yml

no changes added to commit (use "git add" and/or "git commit -a")
jefferson@LocalMachine:~/Final_Exam_Langbid$ git add roles/enterprise/tasks/main.yml
jefferson@LocalMachine:~/Final_Exam_Langbid$ git commit -m "Exam"
[main bc18367] Exam
 1 file changed, 6 deletions(-)
jefferson@LocalMachine:~/Final_Exam_Langbid$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 447 bytes | 223.00 KiB/s, done.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:jangbid13/Final_Exam_Langbid
 523a286..bc18367  main -> main
jefferson@LocalMachine:~/Final_Exam_Langbid$
```

**I added, committed, and pushed it to my github repository.**

5. Make sure to show evidence of input (codes) process (codes successfully running) and output (evidence of installation). Create a word document report for this final exam. For your final exam to be counted, please paste your repository link as an answer in your report. No point will be given if you forgot to paste your repo link.

[jlangbid13/Final\\_Exam\\_Langbid \(github.com\)](https://github.com/jlangbid13/Final_Exam_Langbid)