| | |
|---|---|
| **Name:** Torrecampo, Juan Piolo S. | **Date Performed:** Oct 25, 2022 |
| **Course/Section:** CPE 232 / CPE31S22 | **Date Submitted:** Oct 25, 2022 |
| **Instructor:** Dr. Jonathan Taylar | **Semester and SY:** 1st Sem, 2022 - 2023 |

**Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools**

## 1. Objectives

Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.

## 2. Instructions

1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.
2. Clone the repository and do the following:
   - 2.1. Create an Ansible playbook that does the following with an input of a config.yaml file and arranged Inventory file:
   - 2.2. Install and configure Elastic Stack in separate host
     h) • Install Nagios in one host
   - 2.3. Install Grafana,Prometheus and Influxdb in seperate hosts (Influxdb,Grafana,Prometheus)
   - 2.4. Install Lamp Stack in separate hosts (Httpd + Php,Mariadb)
3. Document all your tasks using this document. Provide proofs of all the ansible playbooks codes and successful installations.
4. Document the push and commit from the local repository to GitHub.
5. Finally, paste also the link of your GitHub repository in the documentation.

## 3. Output (screenshots and explanations)

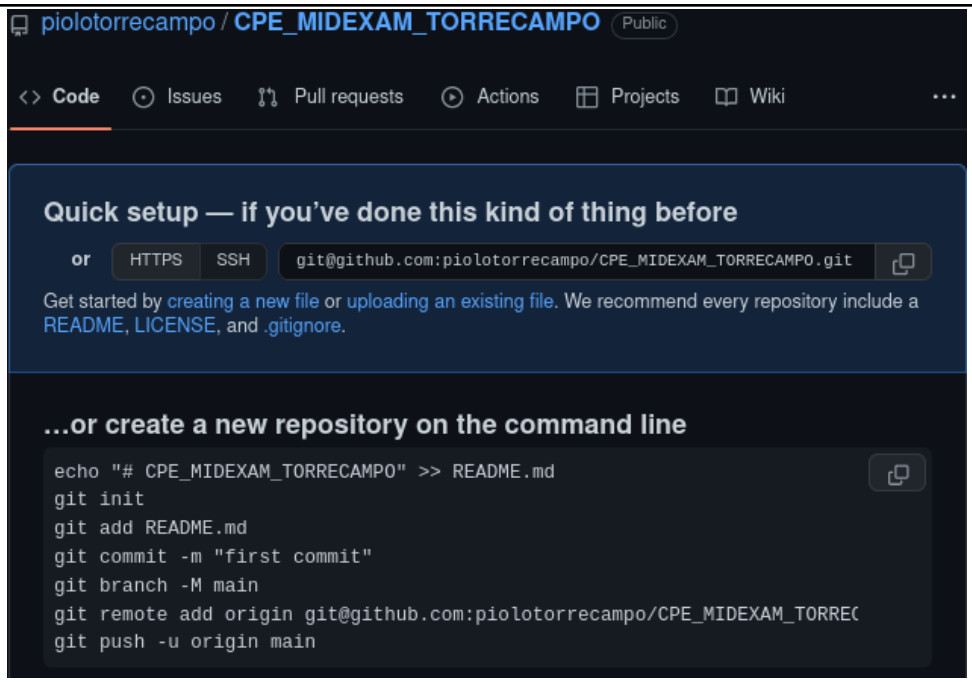



**Figure 1.1. The screenshot above shows the newly created repository in Github.**

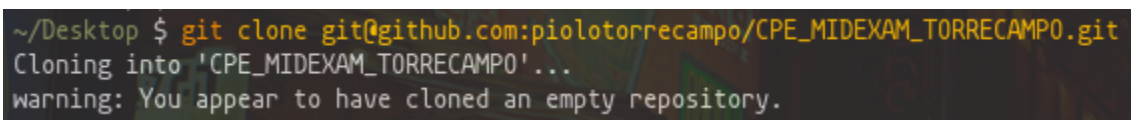



**Fig 1.2. Cloning the repository.**

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) $ tree
.
├── ansible.cfg
├── config.yml
├── files
│   ├── grafana.repo
│   ├── influxdb.repo
│   ├── kibana.repo
│   ├── logstash.repo
│   └── prometheus.service
├── inventory
└── roles
    ├── elk_centos
    │   └── tasks
    │       └── main.yml
    ├── elk_ubuntu
    │   └── tasks
    │       └── main.yml
    ├── igp_centos
    │   └── tasks
    │       └── main.yml
    ├── igp_ubuntu
    │   └── tasks
```

```
    ├── ls_centos
    │   └── tasks
    │       └── main.yml
    ├── ls_ubuntu
    │   ├── tasks
    │   │   └── main.yml
    │   └── tasks.main.yml
    └── nagios_centos
        └── tasks
            └── main.yml

16 directories, 16 files
```

**Figure 1.3. The screenshot above shows the file structure of the ansible.**

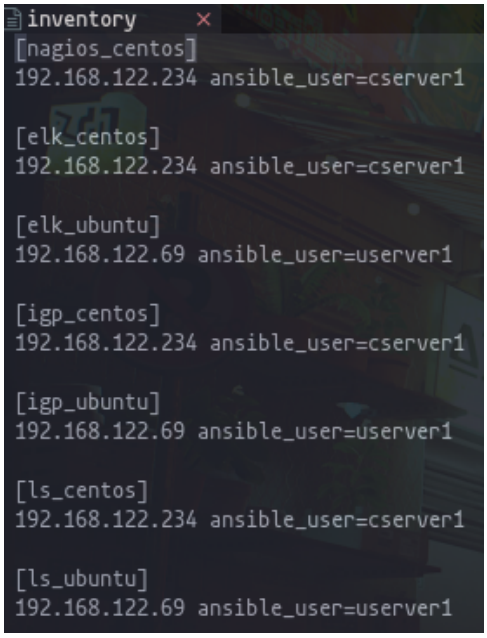
```
ansible.cfg
[defaults]

inventory = inventory
host_key_checking = False

deprecation_warnings = False

private_key_file = ~/.ssh/id_rsa
```

**Figure 1.4. Editing the asible.cfg file.**



```
inventory
[nagios_centos]
192.168.122.234 ansible_user=cserver1

[elk_centos]
192.168.122.234 ansible_user=cserver1

[elk_ubuntu]
192.168.122.69 ansible_user=userver1

[igp_centos]
192.168.122.234 ansible_user=cserver1

[igp_ubuntu]
192.168.122.69 ansible_user=userver1

[ls_centos]
192.168.122.234 ansible_user=cserver1

[ls_ubuntu]
192.168.122.69 ansible_user=userver1
```

**Figure 1.5. Configuring the inventory files to support Ubuntu and Centos.**

config.yml

```yaml
---

- hosts: all
  become: true
  pre_tasks:

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

  - name: Update and upgrade remote CentOS server
    dnf:
      update_cache: yes
      name:  "*"
      state: latest
    when: ansible_distribution == "CentOS"

  - name: Installing installations dependencies
    apt:
      name:
        - wget
      state: latest
    when: ansible_distribution == "Ubuntu"

  - name: Dpkg fixing in Ubuntu Servers
    shell: |
      dpkg --configure -a
    when: ansible_distribution == "Ubuntu"

  - name: Update and upgrade remote in Ubuntu servers
    apt:
      update_cache: yes
      upgrade: yes
    when: ansible_distribution == "Ubuntu"

# Nagios Centos
- hosts: nagios_centos
  tags: nagios
  become: true
  roles:
    - nagios_centos

# ELK Centos
- hosts: elk_centos
  tags: elk
  become: true
  roles:
```

```yaml
  roles:
    - elk_centos

# ELK Ubuntu
- hosts: elk_ubuntu
  tags: elk
  become: true
  roles:
    - elk_ubuntu
```

```
# IGP (Influxdb, Grafana, Prometheus) Centos
- hosts: igp_centos
  tags: igp
  become: true
  roles:
    - igp_centos

# IGP (Influxdb, Grafana, Prometheus) Ubuntu
- hosts: igp_ubuntu
  tags: igp
  become: true
  roles:
    - igp_ubuntu

# Lamp Stack (Centos)
- hosts: ls_centos
  tags: ls
  become: true
  roles:
    - ls_centos

# Lamp Stack (Ubuntu)
- hosts: ls_ubuntu
  tags: ls
  become: true
  roles:
    - ls_ubuntu
```

**Figure 1.6. Contents of config.yml.**

| INSTALLATION OF APPLICATIONS | |
|---|---|
| **Nagios Installation (Centos Server)** | |
| **Script** | **Output** |

```yaml
- 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
    state: directory

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

- name: Compiling, installing, and adding users and groups in nagios
  shell: |
    cd ~/nagios/nagioscore-**
    ./configure
```

```
~/D/CPE_MIDEXM_1QBRECAFO (main|✓) [2]$ ansible-playbook --ask-become-pass config.yml --tags "nagios"
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.09]
ok: [192.168.122.234]

PLAY [nagios_centos] ***********************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.234]

TASK [nagios_centos : Installing nagios dependecies and libraries] *************
ok: [192.168.122.234]

TASK [nagios_centos : Install passlib python package] **************************
ok: [192.168.122.234]

TASK [nagios_centos : Creating a directory (where the downloaded files will be stored)] ***
ok: [192.168.122.234]

TASK [nagios_centos : Downloading and extracting Nagios] ***********************
ok: [192.168.122.234]

TASK [nagios_centos : Compiling, installing, and adding users and groups in nagios] ***
changed: [192.168.122.234]

TASK [nagios_centos : Downloading and extracting Nagios plugins] ***************
ok: [192.168.122.234]

TASK [nagios_centos : Compiling and installing plugins] ************************
changed: [192.168.122.234]

TASK [nagios_centos : Add a user to a password file and ensure permissions are set] ***
ok: [192.168.122.234]

TASK [nagios_centos : Making sure that nagios is started and enabled] **********
changed: [192.168.122.234]

TASK [nagios_centos : Making sure that httpd is started and enabled] ***********
changed: [192.168.122.234]

PLAY [elk_centos] **************************************************************

PLAY [elk_ubuntu] **************************************************************

PLAY [igp_centos] **************************************************************

PLAY [igp_ubuntu] **************************************************************

PLAY [ls_centos] ***************************************************************

PLAY [ls_ubuntu] ***************************************************************

PLAY RECAP *********************************************************************
192.168.122.234            : ok=12   changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
192.168.122.09             : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
```

## ELK (Centos Server)

| Script | Output |
|---|---|

```yaml
# KIBANA
- name: Downloading and installing public signing key
  rpm_key:
    state: present
    key: https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Adding kibana to the RPM repository
  copy:
    src: kibana.repo
    dest: /etc/yum.repos.d/kibana.repo
    owner: root
    group: root
    mode: 777

- name: Updating the repository once again
  yum:
    name:
      - kibana
    state: latest

- name: Opening port for elastic search
  firewalld:
    port: 5601/tcp
    zone: public
    permanent: yes
    state: enabled

- name: Making sure that Kibana is started and enabled
  service:
    name: kibana
    state: restarted
    enabled: true

# LOGSTASH
- name: Downloading and installing public signing key
  rpm_key:
    state: present
    key: https://artifacts.elastic.co/GPG-KEY-elasticsearch

- name: Creeating a repo file for Logstash
  copy:
    src: logstash.repo
    dest: /etc/yum.repos.d/logstash.repo
    owner: root
    group: root
    mode: 0777

- name: Updating repo
  dnf:
    update_cache: yes
```

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) [2]$ ansible-playbook --ask-become-pass config.yml --tags "elk"
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.09]
ok: [192.168.122.234]

PLAY [nagios_centos] ***********************************************************

PLAY [elk_centos] **************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.234]

TASK [elk_centos : Downloading and installing public signing key] **************
ok: [192.168.122.234]

TASK [elk_centos : Adding kibana to the RPM repository] ************************
ok: [192.168.122.234]

TASK [elk_centos : Updating the repository once again] *************************
ok: [192.168.122.234]

TASK [elk_centos : Opening port for elastic search] ****************************
ok: [192.168.122.234]

TASK [elk_centos : Making sure that Kibana is started and enabled] *************
changed: [192.168.122.234]

TASK [elk_centos : Downloading and installing public signing key] **************
ok: [192.168.122.234]

TASK [elk_centos : Creeating a repo file for Logstash] *************************
ok: [192.168.122.234]

TASK [elk_centos : Updating repo] **********************************************
ok: [192.168.122.234]

TASK [elk_centos : Installing Logstash and its dependencies] *******************
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "Failed to download packages: Cannot download 8.4
.3/logstash-8.4.3-x86_64.rpm: All mirrors were tried", "results": []}

PLAY RECAP *********************************************************************
192.168.122.234            : ok=10   changed=1    unreachable=0    failed=1    skipped=0    rescued=0    ignored
=0
192.168.122.09             : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
```

```
main.yml
  dnf:
    name:
      - logstash
    state: latest

- name: Opening port for Logstash
  shell: |
    sudo firewall-cmd --permanent --zone=public --add-port=9600/tcp
    sleep 10
    sudo firewall-cmd --reload

- name: Making sure that logstash is stared and enabled
  service:
    name: logstash
    state: restarted
    enabled: true

# ELASTIC SEARCH
- name: Downloading the source file of elastic search
  get_url:
    url: https://artifacts.elastic.co/downloads/elasticsearch/elasti
    dest: /tmp/elasticsearch-8.4.3-x86_64.rpm

- name: Installing elasticsearch
  yum:
    name: /tmp/elasticsearch-8.4.3-x86_64.rpm
    state: present

- name: Enabling elastic search service
  service:
    name: elasticsearch
    enabled: yes

- name: Modifying service file
  replace:
    path: /usr/lib/systemd/system/elasticsearch.service
    regexp: "TimeoutStartSec=75"
    replace: "TimeoutStartSec=300"

- name: Opening port for elastic search
  shell: |
    sudo firewall-cmd --permanent --zone=public --add-port=9200/tcp
    sleep 10
    sudo firewall-cmd --reload

- name: Enabling elastic search service
  shell: |
    systemctl enable elasticsearch.service
    sleep 10
    systemctl start elasticsearch.service
```

## ELK (Centos Server)

| Script | Output |
|---|---|

```
main.yml
# ELASTIC SEARCH
- name: Downloading and adding GPG keys
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/k

- name: Installing dependecies
  apt:
    name:
      -  apt-transport-https
    state: latest

- name: Adding es to the repo
  shell: |
    echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/p

- name: Updating the repo and installing es
  apt:
    name: elasticsearch
    update_cache: true

- name: Reloading daemon
  shell: |
    sudo /bin/systemctl daemon-reload

- name: Starting and enabling the process
  service:
    name: elasticsearch
    state: lastest
    enabled: true

# LOGSTASH
- name: Downlaoding the public signing key and adding to the repo
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

- name: Donwloading dependecies
  apt:
    name:
      - apt-transport-https
    state: present

- name: Saving repo definition
  shell: |
    echo "deb https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee -a /etc/apt/source

- name: Updatin and installing Logstash
  apt:
    name:
      - logstash
```

```
- name: Updatin and installing Logstash
  apt:
    name:
      - logstash
    update_cache: yes


# KIBANA
- name: Downlaoding the public signing key and adding to the repo
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

- name: Donwloading dependecies
  apt:
    name:
      - apt-transport-https
    state: present

- name: Saving repo definition
  shell: |
    echo "deb https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee -a /etc/apt/source

- name: Updatin and installing Logstash
  apt:
    name:
      - kibana
    update_cache: yes

- name: Reloading daemon
  shell: |
    sudo /bin/systemctl daemon-reload

- name: Starting and enabling the process
  service:
    name: kibana
    state: lastest
    enabled: true
```

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) [2]$ ansible-playbook --ask-become-pass config.yml --tags "elk"
BECOME password:

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

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.09]
ok: [192.168.122.234]

PLAY [nagios_centos] ***********************************************************

PLAY [elk_centos] **************************************************************

TASK [Gathering Facts] *********************************************************
ok: [192.168.122.234]

TASK [elk_centos : Downloading and installing public signing key] **************
ok: [192.168.122.234]

TASK [elk_centos : Adding kibana to the RPM repository] ************************
ok: [192.168.122.234]

TASK [elk_centos : Updating the repository once again] *************************
ok: [192.168.122.234]

TASK [elk_centos : Opening port for elastic search] ****************************
ok: [192.168.122.234]

TASK [elk_centos : Making sure that Kibana is started and enabled] *************
changed: [192.168.122.234]

TASK [elk_centos : Downloading and installing public signing key] **************
ok: [192.168.122.234]

TASK [elk_centos : Creeating a repo file for Logstash] *************************
ok: [192.168.122.234]

TASK [elk_centos : Updating repo] **********************************************
ok: [192.168.122.234]

TASK [elk_centos : Installing Logstash and its dependencies] *******************
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "Failed to download packages: Cannot download 8.4
.3/logstash-8.4.3-x86_64.rpm: All mirrors were tried", "results": []}

PLAY RECAP *********************************************************************
192.168.122.234            : ok=10   changed=1    unreachable=0    failed=1    skipped=0    rescued=0    ignored
=0
192.168.122.09             : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
```

## ELK (Ubuntu Server)

| Script | Output |
|---|---|

```
# ELASTIC SEARCH
- name: Downloading and adding GPG keys
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/k

- name: Installing dependecies
  apt:
    name:
      -  apt-transport-https
    state: latest

- name: Adding es to the repo
  shell: |
    echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/p

- name: Updating the repo and installing es
  apt:
    name: elasticsearch
    update_cache: true

- name: Reloading daemon
  shell: |
    sudo /bin/systemctl daemon-reload

- name: Starting and enabling the process
  service:
    name: elasticsearch
    state: lastest
    enabled: true

# LOGSTASH
- name: Downlaoding the public signing key and adding to the repo
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

- name: Donwloading dependecies
  apt:
    name:
      - apt-transport-https
    state: present

- name: Saving repo definition
  shell: |
    echo "deb https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee -a /etc/apt/source

- name: Updatin and installing Logstash
  apt:
    name:
      - logstash
```

```
    update_cache: yes

# KIBANA
- name: Downlaoding the public signing key and adding to the repo
  shell: |
    wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

- name: Donwloading dependecies
  apt:
    name:
      - apt-transport-https
    state: present

- name: Saving repo definition
  shell: |
    echo "deb https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee -a /etc/apt/source

- name: Updatin and installing Logstash
  apt:
    name:
      - kibana
    update_cache: yes

- name: Reloading daemon
  shell: |
    sudo /bin/systemctl daemon-reload

- name: Starting and enabling the process
  service:
    name: kibana
    state: lastest
    enabled: true
```

## IGP  - Influxdb, Grafana, Prometheus - (Centos)

| Script | Output |
|---|---|

```yaml
# PROMETHEUS
- name: Creating a directory (where the downloaded files will be stored)
  tags: directory
  file:
    path: ~/prometheus
    state: directory

- name: Downloading and extracting Prometheus
  tags: source
  unarchive:
    src: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-am
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Stopping the service
  service:
    name: prometheus
    state: stopped

- name: Adding the Prometheus executables to a PATH
  tags: executables
  shell: |
    cd ~/prometheus/prometheus*
    cp -r . /usr/local/bin/prometheus

- name: Copying the Prometheus service file
  tags: servicefile
  copy:
    src: prometheus.service
    dest: /etc/systemd/system/
    owner: root
    group: root
    mode: 777

- name: Making sure that Prometheus is started and enabled
  tags: serviceon
  service:
    name: prometheus
    state: restarted
    enabled: true

# GRAFANA
- name: Adding Grafana to the repository
  copy:
    src: grafana.repo
    dest: /etc/yum.repos.d/grafana.repo
```

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) $ ansible-playbook --ask-become-pass config.yml --tags "igp"
BECOME password:

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

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.122.09]
ok: [192.168.122.234]

PLAY [nagios_centos] ***********************************************************************************

PLAY [elk_centos] **************************************************************************************

PLAY [elk_ubuntu] **************************************************************************************

PLAY [igp_centos] **************************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.122.234]

TASK [igp_centos : Creating a directory (where the downloaded files will be stored)] *******************
changed: [192.168.122.234]

TASK [igp_centos : Downloading and extracting Prometheus] **********************************************
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "Failure downloading https://github.com/prometheu
s/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-amd64.tar.gz, ('The read operation timed out',)"}

PLAY RECAP *********************************************************************************************
192.168.122.234            : ok=3    changed=1    unreachable=0    failed=1    skipped=0    rescued=0    ignored
=0
192.168.122.09             : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
```

## IGP - Influxdb, Grafana, Prometheus (Ubuntu)

### Script

```yaml
# PROMETHEUS
- name: Creating a directory (where the downloaded files will be stored)
  tags: directory
  file:
    path: ~/prometheus
    state: directory

- name: Downloading and extracting Prometheus
  tags: source
  unarchive:
    src: https://github.com/prometheus/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-am
    dest: ~/prometheus
    remote_src: yes
    mode: 0777
    owner: root
    group: root

- name: Stopping the service
  service:
    name: prometheus
    state: stopped

- name: Adding the Prometheus executables to a PATH
  tags: executables
  shell: |
    cd ~/prometheus/prometheus*
    cp -r . /usr/local/bin/prometheus
- name: Copying the Prometheus service file
  tags: servicefile
  copy:
    src: prometheus.service
    dest: /etc/systemd/system/
    owner: root
    group: root
    mode: 777

- name: Making sure that httpd is started and enabled
  tags: serviceon
  service:
    name: prometheus
    state: restarted
    enabled: true

# GRAFANA
- name: Installing dependencies
  apt:
    name:
      - apt-transport-https
      - software-properties-common
      - wget
```

### Output

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) $ ansible-playbook --ask-become-pass config.yml --tags "igp"
BECOME password:

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

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.122.09]
ok: [192.168.122.234]

PLAY [nagios_centos] ***********************************************************************************

PLAY [elk_centos] **************************************************************************************

PLAY [elk_ubuntu] **************************************************************************************

PLAY [igp_centos] **************************************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [192.168.122.234]

TASK [igp_centos : Creating a directory (where the downloaded files will be stored)] *******************
changed: [192.168.122.234]

TASK [igp_centos : Downloading and extracting Prometheus] **********************************************
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "Failure downloading https://github.com/prometheu
s/prometheus/releases/download/v2.39.1/prometheus-2.39.1.linux-amd64.tar.gz, ('The read operation timed out',)"}

PLAY RECAP *********************************************************************************************
192.168.122.234            : ok=3    changed=1    unreachable=0    failed=1    skipped=0    rescued=0    ignored
=0
192.168.122.09             : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored
=0
```

```
main.yml
    dest: /etc/yum.repos.d/grafana.repo
    owner: root
    group: root
    mode: 777

- name: Installing Grafana
  yum:
    name:
      - grafana
    state: latest

- name: Reloading the daemon
  shell: |
    sudo systemctl daemon-reload

- name: Sarting and enabling the service
  service:
    name: grafana-server
    state: restarted
    enabled: true

# INFLUXD
- name: Adding influxdb in repository
  copy:
    scr: influxdb.repo
    dest: /etc/yum.repos.d/influxdb.repo
    owner: root
    group: root
    mode: 777

- name: Installing grafana
  yum:
    name:
      - grafana
    state: latest

- name: Reloading the daemon
  shell: |
    sudo systemctl daemon-reload

- name: Sarting and enabling the service
  service:
    name: influxdb
    state: restarted
    enabled: true

- name: Opening ports
  shell: |
    firewall-cmd --permanent --zone=public --add-port=8086/tcp
    sleep 5
```

```
    state: latest

- name: Adding Grafana Repo
  shell: |
    echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://packages.grafana.com/oss/deb stable

- name: Updating the repo and isntalling grafana
  apt:
    name:
      - grafana
    update_cache: true
    state: lastest

- name: Reloading the daemon
  shell: |
    sudo systemctl daemon-reload

- name: Sarting and enabling the service
  service:
    name: grafana-server
    state: restarted
    enabled: true

# INFLUXDB
- name: Adding influxdb in the repo
  shell: |
    echo "deb https://repos.influxdata.com/ubuntu focal stable" | sudo tee /etc/apt/sources.list.d/inf

- name: Updating and installing influxdb
  apt:
    name:
      - influxdb
    update_cache: true

- name: Starting and enabling the service
  service:
    name: influxdb
    state: restarted
    enabled: true
```

## LP - Lamp Stack (Centos)

| Script | Output |
|---|---|

```yaml
main.yml
- name: Installing apache
  yum:
    name:
      - httpd
      - mariadb-server
      - mariadb
      - php
      - php-mysql
    state: latest

- name: Starting the services
  service:
    name:
      - httpd
      - mariadb
      - apache2
    state: restarted
    enabled: true

- name: Opening needed ports
  shell: |
    sudo firewall-cmd --permanent --zone=public --add-service=http
    sudo firewall-cmd --permanent --zone=public --add-service=https
    sudo firewall-cmd --reload
```

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✔) $ ansible-playbook --ask-become-p
ass config.yml --tags "ls"
BECOME password:

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

TASK [Gathering Facts] ***********************************************
***********
ok: [192.168.122.69]
ok: [192.168.122.234]

PLAY [nagios_centos] *************************************************
***********

PLAY [elk_centos] ****************************************************
***********

PLAY [elk_ubuntu] ****************************************************
***********

PLAY [igp_centos] ****************************************************
***********

PLAY [igp_ubuntu] ****************************************************
***********

PLAY [ls_centos] *****************************************************
***********

TASK [Gathering Facts] ***********************************************
***********
ok: [192.168.122.234]

TASK [ls_centos : Installing apache] *********************************
***********
ok: [192.168.122.234]

TASK [ls_centos : Starting the services] *****************************
***********
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "This
module does not currently support using glob patterns, found '[' in s
ervice name: ['httpd', 'mariadb', 'apache2']"}

PLAY RECAP ***********************************************************
***********
192.168.122.234            : ok=3    changed=0    unreachable=0    fa
iled=1    skipped=0    rescued=0    ignored=0
192.168.122.69             : ok=1    changed=0    unreachable=0    fa
iled=0    skipped=0    rescued=0    ignored=0
```

## LP - Lamp Stack (Ubuntu)

| Script | Output |
|---|---|

```yaml
name: Installing depedncies
apt:
  name:
    - apache2
  state: latest

- name: Starting the services
service:
  name:
    - apache2
    - mariadb
  states: restarted
  enabled: true
```

```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✓) $ ansible-playbook --ask-become-p
ass config.yml --tags "ls"
BECOME password:

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

TASK [Gathering Facts] ***********************************************
***********
ok: [192.168.122.69]
ok: [192.168.122.234]

PLAY [nagios_centos] *************************************************
***********

PLAY [elk_centos] ****************************************************
***********

PLAY [elk_ubuntu] ****************************************************
***********

PLAY [igp_centos] ****************************************************
***********

PLAY [igp_ubuntu] ****************************************************
***********

PLAY [ls_centos] *****************************************************
***********

TASK [Gathering Facts] ***********************************************
***********
ok: [192.168.122.234]

TASK [ls_centos : Installing apache] *********************************
***********
ok: [192.168.122.234]

TASK [ls_centos : Starting the services] *****************************
***********
fatal: [192.168.122.234]: FAILED! => {"changed": false, "msg": "This
module does not currently support using glob patterns, found '[' in s
ervice name: ['httpd', 'mariadb', 'apache2']"}

PLAY RECAP ***********************************************************
***********
192.168.122.234            : ok=3    changed=0    unreachable=0    fa
iled=1    skipped=0    rescued=0    ignored=0
192.168.122.69             : ok=1    changed=0    unreachable=0    fa
iled=0    skipped=0    rescued=0    ignored=0
```

# T.I.P. Faculty Performance Evaluation by the Students (1st Semester, S.Y. 2022-2023, Modular Group 4-part 2)

Your response has been recorded.

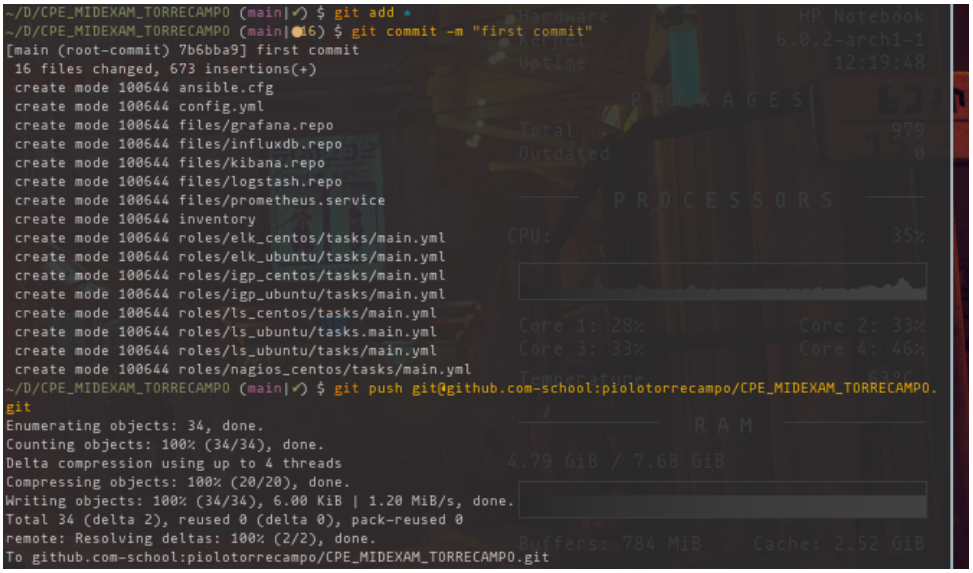
Edit your response

Submit another response

This form was created inside of Technological Institute of the Philippines. Report Abuse

Google Forms

**Figure 1.7. Accomplishing the TIP evaluation.**



```
~/D/CPE_MIDEXAM_TORRECAMPO (main|✔) $ git add *
~/D/CPE_MIDEXAM_TORRECAMPO (main|●16) $ git commit -m "first commit"
[main (root-commit) 7b6bba9] first commit
 16 files changed, 673 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 config.yml
 create mode 100644 files/grafana.repo
 create mode 100644 files/influxdb.repo
 create mode 100644 files/kibana.repo
 create mode 100644 files/logstash.repo
 create mode 100644 files/prometheus.service
 create mode 100644 inventory
 create mode 100644 roles/elk_centos/tasks/main.yml
 create mode 100644 roles/elk_ubuntu/tasks/main.yml
 create mode 100644 roles/igp_centos/tasks/main.yml
 create mode 100644 roles/igp_ubuntu/tasks/main.yml
 create mode 100644 roles/ls_centos/tasks/main.yml
 create mode 100644 roles/ls_ubuntu/tasks.main.yml
 create mode 100644 roles/ls_ubuntu/tasks/main.yml
 create mode 100644 roles/nagios_centos/tasks/main.yml
~/D/CPE_MIDEXAM_TORRECAMPO (main|✔) $ git push git@github.com-school:piolotorrecampo/CPE_MIDEXAM_TORRECAMPO.
git
Enumerating objects: 34, done.
Counting objects: 100% (34/34), done.
Delta compression using up to 4 threads
Compressing objects: 100% (20/20), done.
Writing objects: 100% (34/34), 6.00 KiB | 1.20 MiB/s, done.
Total 34 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To github.com-school:piolotorrecampo/CPE_MIDEXAM_TORRECAMPO.git
```

**Figure 1.8. Pushing the files to Github.**

**GitHub link:** https://github.com/piolotorrecampo/CPE_MIDEXAM_TORRECAMPO.git

**Conclusions:**

This midterm exam achieved its goal to create a workflow that installs, configures and manages enterprise availability, performance and log monitoring tools using Ansible as an Iac tool. In this state the two servers are almost the same configuration with the enterprise servers that monitor and manage the servers. While making this skill examination program, I realized that a server should have a powerful system unable to run this bunch of services in the background. For system administrators, Ansible is a huge help to them in managing a bunch of servers in small or large geographical areas. Therefore, Ansible is a great tool to manage different system remotely which enable to install and configure applications.

**Honor Pledge:**

*"I affirm that I will not give or receive unauthorized help on this activity and that all will be my own.*