

Name:	Date Performed:
Course/Section:	Date Submitted:
Instructor:	Semester and SY:
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	
<p>1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME.</p> <pre> robin@robin-workstation:~\$ git clone git@github.com:Patrickcruz14/CPE232_Patrickcruz.git Cloning into 'CPE232_Patrickcruz'... The authenticity of host 'github.com (4.237.22.38)' can't be established. ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvC0qU. This key is not known by any other names. Are you sure you want to continue connecting (yes/no/[fingerprint])? yes Warning: Permanently added 'github.com' (ED25519) to the list of known hosts. remote: Enumerating objects: 6, done. remote: Counting objects: 100% (6/6), done. remote: Compressing objects: 100% (2/2), done. remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0) Receiving objects: 100% (6/6), done. robin@robin-workstation:~\$ git clone git@github.com:Patrickcruz14/CPE_MIDEXAM_CRUZ.git Cloning into 'CPE_MIDEXAM_CRUZ'... remote: Enumerating objects: 3, done. remote: Counting objects: 100% (3/3), done. remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0) Receiving objects: 100% (3/3), done. </pre> <p>2. Clone the repository and do the following:</p> <ol style="list-style-type: none"> 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 hosts (Elastic Search, Kibana, Logstash) • 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) <p>(elasticsearch main.yml)</p>	

None

- name: Apply configuration from config.yaml
hosts: all
gather_facts: yes
vars_files:
 - config.yamltasks:
 - debug: msg="Running plays defined below"

- name: Install Elasticsearch
hosts: elasticsearch
roles:
 - elasticsearch

- name: Install Kibana
hosts: kibana
roles:
 - kibana

- name: Install Logstash
hosts: logstash
roles:
 - logstash

```
- name: Install Nagios (server)
  hosts: nagios
  roles:
    - nagios

- name: Install LAMP web (apache + php)
  hosts: web
  roles:
    - lamp_web

- name: Install MariaDB
  hosts: db
  roles:
    - lamp_db
```

None

```
- name: Add Elastic GPG key (Debian)
  when: ansible_os_family == 'Debian'
  apt_key:
    url:
      https://artifacts.elastic.co/GPG-KEY-elasticsearch
```

```
    state: present

- name: Add Kibana repo (Debian)
  when: ansible_os_family == 'Debian'
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/{{
elastic_version.split('.')[0] }}.x/apt stable main"
    state: present

- name: Add Kibana repo (RedHat/CentOS)
  when: ansible_os_family == 'RedHat'
  yum_repository:
    name: elastic-{{ elastic_version.split('.')[0] }}-x
    description: Elastic repository
    baseurl: "https://artifacts.elastic.co/packages/{{
elastic_version.split('.')[0] }}.x/yum"
    gpgcheck: yes
    gpgkey:
https://artifacts.elastic.co/GPG-KEY-elasticsearch
    enabled: yes

- name: Install kibana
  package:
    name: kibana
    state: present
```

```
- name: Configure kibana.yml
  copy:
    dest: /etc/kibana/kibana.yml
    content: |
      server.port: {{ kibana_server_port }}
      server.host: "0.0.0.0"
      elasticsearch.hosts: ["http://{{ hosts.elasticsearch
}}:9200"]
  notify: restart kibana

- name: enable and start kibana
  systemd:
    name: kibana
    enabled: yes
    state: started
```

None

```
- name: Add Elastic GPG key (Debian)
  when: ansible_os_family == 'Debian'
  apt_key:
```

```

    url:
https://artifacts.elastic.co/GPG-KEY-elasticsearch

    state: present

- name: Add Logstash repo (Debian)
  when: ansible_os_family == 'Debian'
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/{{
elastic_version.split('.')[0] }}.x/apt stable main"
    state: present

- name: Install logstash
  package:
    name: logstash
    state: present

- name: Create basic pipeline config
  copy:
    dest: "{{ logstash_pipeline_dir }}/simple.conf"
    content: |
        input { tcp { port => 5000 codec => json_lines } }
        output { elasticsearch { hosts => ["{{
hosts.elasticsearch }}:9200"] } }
  notify: restart logstash

```

```
- name: Ensure logstash enabled and running
  systemd:
    name: logstash
    enabled: yes
    state: started
```

None

```
---
- name: Add Elastic GPG key (Debian)
  when: ansible_os_family == 'Debian'
  apt_key:
    url:
      https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present

- name: Add Logstash repo (Debian)
  when: ansible_os_family == 'Debian'
  apt_repository:
    repo: "deb https://artifacts.elastic.co/packages/{{
      elastic_version.split('.')[0] }}.x/apt stable main"
    state: present
```

```
- name: Install logstash
  package:
    name: logstash
    state: present

- name: Create basic pipeline config
  copy:
    dest: "{{ logstash_pipeline_dir }}/simple.conf"
    content: |
      input { tcp { port => 5000 codec => json_lines } }
      output { elasticsearch { hosts => ["{{
hosts.elasticsearch }}:9200"] } }
    notify: restart logstash

- name: Ensure logstash enabled and running
  systemd:
    name: logstash
    enabled: yes
    state: started
```

None

```
- name: Add Elastic GPG key (Debian)
```



```

    when: ansible_os_family == 'Debian'
    apt_key:
        url:
https://artifacts.elastic.co/GPG-KEY-elasticsearch
        state: present

- name: Add Logstash repo (Debian)
    when: ansible_os_family == 'Debian'
    apt_repository:
        repo: "deb https://artifacts.elastic.co/packages/{{
elastic_version.split('.')[0] }}.x/apt stable main"
        state: present

- name: Install logstash
    package:
        name: logstash
        state: present

- name: Create basic pipeline config
    copy:
        dest: "{{ logstash_pipeline_dir }}/simple.conf"
        content: |
            input { tcp { port => 5000 codec => json_lines } }
            output { elasticsearch { hosts => ["{{
hosts.elasticsearch }}:9200" ] } }

```

```
notify: restart logstash
```

```
- name: Ensure logstash enabled and running
```

```
systemd:
```

```
  name: logstash
```

```
  enabled: yes
```

```
  state: started
```

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)

GitHub link:

Conclusions: (link your conclusion from the objective)