

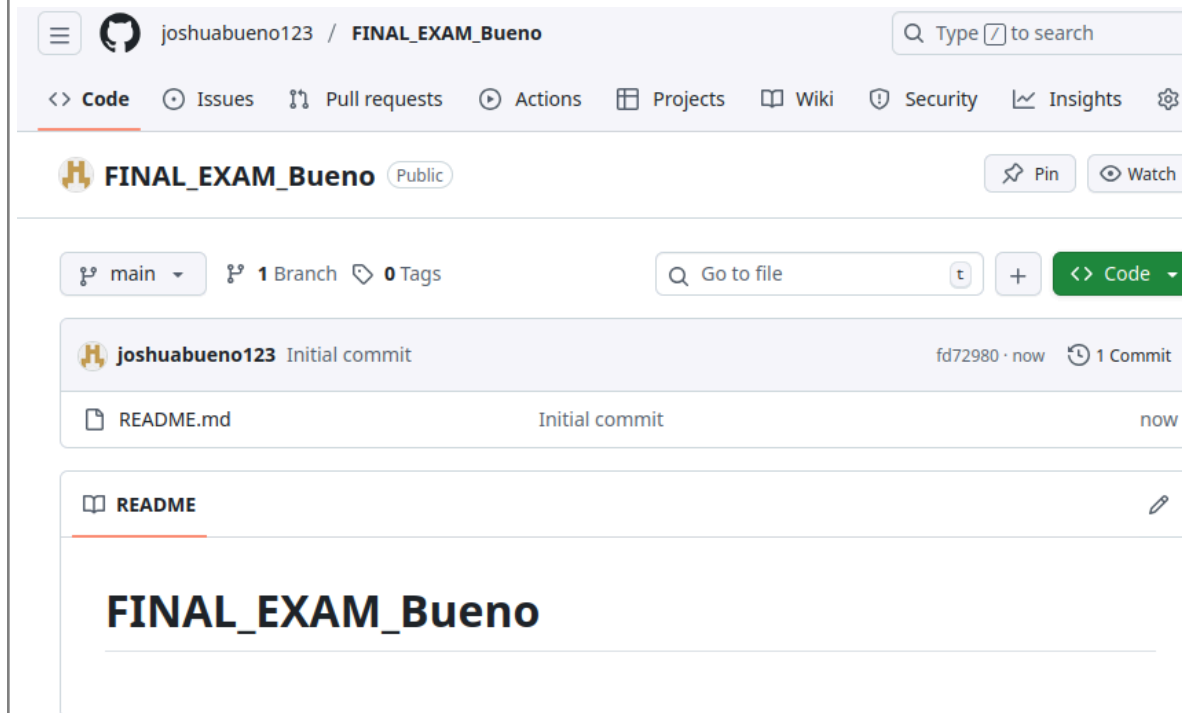
Bueno, Joshua C.	11/14/25
CPE31S2	Engr. Robin Valenzuela

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"



Analysis: I created a repository on my github account to put all the file that I will create in this exam

2. Clone your new repository in your VM

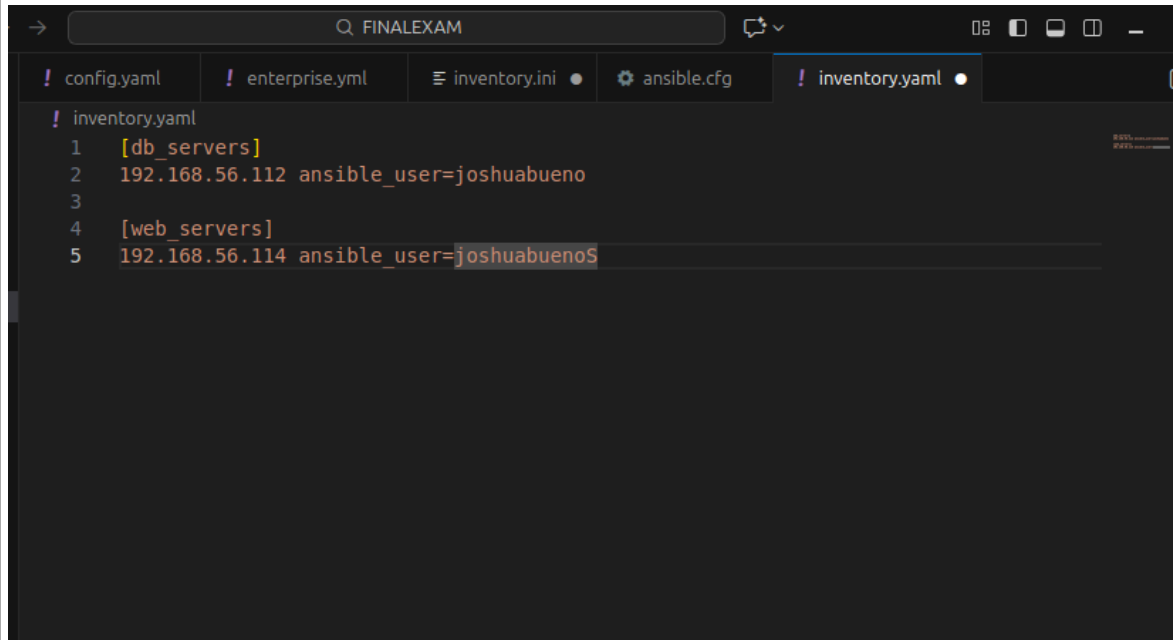
```

s://github.com/joshuabueno123/FINAL_EXAM_Bueno.git
Cloning into 'FINAL_EXAM_Bueno'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pac
-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

```

Analysis: I gitclone my repository that I created to have a backup file if I need and put it on my github account.

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

A screenshot of a code editor window titled 'FINALEXAM'. The editor has several tabs open: 'config.yaml', 'enterprise.yml', 'inventory.ini', 'ansible.cfg', and 'inventory.yaml'. The 'inventory.yaml' tab is active, showing the following content:

```
! inventory.yaml
1  [db_servers]
2  192.168.56.112 ansible_user=joshuabueno
3
4  [web_servers]
5  192.168.56.114 ansible_user=joshuabueno
```

Analysis: I created all the config files and inventory that I will be using in installation like inventory, ansible, and config.yaml

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

## Code:

```
! enterprise.yml
1  ---
2  - name: Setup enterprise service and monitoring
3    hosts: all
4    become: true
5    vars_files:
6      - ./config.yaml    # make sure this file is in the same directory as playbook
7
8    tasks:
9
10     - name: Ensure package manager cache is up-to-date
11       ansible.builtin.package:
12         update_cache: yes
13
14     # 3.1 Enterprise Service: Apache
15     - name: Install Apache web server
16       ansible.builtin.package:
17         name: "{{ 'apache2' if ansible_os_family == 'Debian' else 'httpd' }}"
18         state: present
19
20     - name: Start and enable Apache service
21       ansible.builtin.service:
22         name: "{{ 'apache2' if ansible_os_family == 'Debian' else 'httpd' }}"
23         state: started
24         enabled: yes
25
26     # 3.2 Monitoring Tool: Node Exporter
27     - name: Download Node Exporter binary
28       ansible.builtin.get_url:
29         url: "https://github.com/prometheus/node_exporter/releases/download/v1.8.1"
30         dest: /tmp/node_exporter.tar.gz
31         mode: '0644'
32         when: monitoring_tool is defined and monitoring_tool == "node_exporter"
33
34     - name: Extract Node Exporter
35       ansible.builtin.unarchive:
```

```

35     ansible.builtin.unarchive:
36         src: /tmp/node_exporter.tar.gz
37         dest: /usr/local/bin/
38         remote_src: yes
39         extra_opts: [--strip-components=1]
40     when: monitoring_tool is defined and monitoring_tool == "node_exporter"
41
42     - name: Create systemd service for Node Exporter
43       ansible.builtin.copy:
44         dest: /etc/systemd/system/node_exporter.service
45         content: |
46             [Unit]
47             Description=Prometheus Node Exporter
48             After=network.target
49
50             [Service]
51             ExecStart=/usr/local/bin/node_exporter
52             User=nobody
53             Group=nogroup
54
55             [Install]
56             WantedBy=multi-user.target
57     when: monitoring_tool is defined and monitoring_tool == "node_exporter"
58
59     - name: Start and enable Node Exporter
60       ansible.builtin.systemd:
61         name: node_exporter
62         state: started
63         enabled: yes
64     when: monitoring_tool is defined and monitoring_tool == "node_exporter"

```

```

64     when: monitoring_tool is defined and monitoring_tool == "node_exporter"
65
66     # 4.4 MOTD Message
67     - name: Update MOTD
68       ansible.builtin.copy:
69         dest: /etc/motd
70         content: "Ansible Managed by {{ motd_user | default('unknown_user') }}\n"
71         owner: root
72         group: root
73         mode: '0644'
74

```

Analysis: This is the code that I created while installing the enterprise service in Debian and CentOS.

Output:  
Debian

```

TASK [Gathering Facts] *****
ok: [192.168.56.112]

TASK [Package Manager Cache] *****
changed: [192.168.56.112]

TASK [Install Apache web server] *****
ok: [192.168.56.112]

TASK [Start and enable Apache service] *****
ok: [192.168.56.112]

TASK [Download Node Exporter binary] *****
skipping: [192.168.56.112]

TASK [Extract Node Exporter] *****
skipping: [192.168.56.112]

TASK [Create systemd service for Node Exporter] *****
skipping: [192.168.56.112]

TASK [Start and enable Node Exporter] *****
skipping: [192.168.56.112]

TASK [Update MOTD] *****
ok: [192.168.56.112]

PLAY RECAP *****
192.168.56.112      : ok=5    changed=1    unreachable=0    failed=0    skipped=4    rescued=0    ignored=0

```

**Analysis:** It shows in the image that all the tasks was successfully initiated and installed in Debian server.

## CENTOS:

```

TASK [Ensure package manager cache is up-to-date] *****
ok: [192.168.56.114]

TASK [Install Apache web server (httpd)] *****
ok: [192.168.56.114]

TASK [Start and enable Apache service] *****
changed: [192.168.56.114]

TASK [Download Node Exporter binary] *****
skipping: [192.168.56.114]

TASK [Extract Node Exporter] *****
skipping: [192.168.56.114]

TASK [Create systemd service for Node Exporter] *****
skipping: [192.168.56.114]

TASK [Start and enable Node Exporter] *****
skipping: [192.168.56.114]

TASK [Update MOTD] *****
changed: [192.168.56.114]

TASK [Clean up Node Exporter tar file] *****
skipping: [192.168.56.114]

PLAY RECAP *****
192.168.56.114      : ok=5    changed=2    unreachable=0    failed=0    skipped=5    rescued=0    ignored=0

```

**Analysis:** Based on the output, it successfully downloaded the enterprise in both Debian and CentOS

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)

**Debian:**

```

PLAY [Install and configure Node Exporter on Debian servers] *****
TASK [Gathering Facts] *****
ok: [192.168.56.112]
TASK [Update APT cache] *****
changed: [192.168.56.112]
TASK [Install required tools on Debian] *****
ok: [192.168.56.112]
TASK [Download Node Exporter binary for Debian] *****
changed: [192.168.56.112]
TASK [Extract Node Exporter] *****
changed: [192.168.56.112]
TASK [Create systemd service for Node Exporter] *****
changed: [192.168.56.112]
TASK [Reload systemd] *****
ok: [192.168.56.112]
TASK [Enable and start Node Exporter service] *****
changed: [192.168.56.112]
PLAY RECAP *****
192.168.56.112      : ok=8    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

Analysis: The installation of monitoring tool was successful in Debian as the output shows all the tasks run smoothly

## CENTOS:

```

BECOME password:
PLAY [Install and configure Node Exporter on CentOS servers] *****
TASK [Gathering Facts] *****
ok: [192.168.56.114]
TASK [Install required tools on CentOS] *****
ok: [192.168.56.114]
TASK [Download Node Exporter binary for CentOS] *****
changed: [192.168.56.114]
TASK [Extract Node Exporter] *****
changed: [192.168.56.114]
TASK [Create systemd service for Node Exporter] *****
changed: [192.168.56.114]
TASK [Reload systemd] *****
ok: [192.168.56.114]
TASK [Enable and start Node Exporter service] *****
changed: [192.168.56.114]
PLAY RECAP *****
192.168.56.114      : ok=7    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

Analysis: The installation of monitoring tool was successful also in CentOS VM as the output shows all the tasks run smoothly

## Prometheus Server

Code:

```
! prometheus.yml
1 ---
2 - name: Install and configure Prometheus server
3   hosts: prometheus_server2
4   become: true
5   vars_files:
6   - config.yml
7
8   tasks:
9     # Ensure the /etc/prometheus directory exists
10    - name: Ensure the /etc/prometheus directory exists
11      ansible.builtin.file:
12        path: /etc/prometheus
13        state: directory
14        mode: '0755'
15
16    # Step 1: Install Prometheus on Debian (APT)
17    - name: Install Prometheus on Debian
18      ansible.builtin.apt:
19        update_cache: yes
20        name: prometheus
21        state: present
22      when: ansible_os_family == "Debian"
23
24    # Step 2: Install dependencies on CentOS
25    - name: Install dependencies on CentOS (wget)
26      ansible.builtin.yum:
27        name: wget
28        state: present
29      when: ansible_os_family == "RedHat"
30
31    # Step 3: Download Prometheus binary (CentOS)
32    - name: Download Prometheus binary (CentOS)
33      ansible.builtin.get_url:
34        url: "https://github.com/prometheus/prometheus/releases/download/v2.53.1/p
35        dest: /tmp/prometheus.tar.gz
```

! prometheus.yml

```
35     dest: /tmp/prometheus.tar.gz
36     when: ansible_os_family == "RedHat"
37
38     # Step 4: Extract Prometheus binary (CentOS)
39     - name: Extract Prometheus binary (CentOS)
40       ansible.builtin.unarchive:
41         src: /tmp/prometheus.tar.gz
42         dest: /usr/local/bin/
43         extra_opts: ["--strip-components=1"]
44         remote_src: yes
45       when: ansible_os_family == "RedHat"
46
47     # Step 5: Configure Prometheus to scrape all Node Exporters
48     - name: Configure Prometheus to scrape all Node Exporters
49       ansible.builtin.copy:
50         dest: /etc/prometheus/prometheus.yml
51         content: |
52             global:
53               scrape_interval: 15s
54             scrape_configs:
55               - job_name: "node_exporter"
56                 static_configs:
57                   - targets:
58                       {% for host in groups['all_node_exporter'] %}
59                       - "{{ host }}:9100"
60                       {% endfor %}
61
62     # Step 6: Create Prometheus systemd service (for CentOS)
63     - name: Create Prometheus systemd service (CentOS)
64       ansible.builtin.copy:
65         dest: /etc/systemd/system/prometheus.service
66         content: |
67             [Unit]
68             Description=Prometheus Monitoring
```



```

! prometheus.yml
69     Wants=network-online.target
70     After=network-online.target
71
72     [Service]
73     User=nobody
74     Group=nogroup
75     Type=simple
76     ExecStart=/usr/local/bin/prometheus \
77         --config.file /etc/prometheus/prometheus.yml \
78         --storage.tsdb.path /var/lib/prometheus \
79         --web.console.templates=/usr/local/bin/consoles \
80         --web.console.libraries=/usr/local/bin/console_libraries
81
82     [Install]
83     WantedBy=multi-user.target
84     when: ansible_os_family == "RedHat"
85
86     # Step 7: Reload systemd daemon
87     - name: Reload systemd daemon
88       ansible.builtin.systemd:
89         daemon_reload: yes
90
91     # Step 8: Enable and start Prometheus service
92     - name: Enable and start Prometheus service
93       ansible.builtin.systemd:
94         name: prometheus
95         state: started
96         enabled: yes
97
98     # Step 9: Update MOTD
99     - name: Get current user for MOTD
100       ansible.builtin.command:
101         cmd: whoami
102       register: current_user

```

```

! prometheus.yml
69     Wants=network-online.target
70     After=network-online.target
71
72     [Service]
73     User=nobody
74     Group=nogroup
75     Type=simple
76     ExecStart=/usr/local/bin/prometheus \
77         --config.file /etc/prometheus/prometheus.yml \
78         --storage.tsdb.path /var/lib/prometheus \
79         --web.console.templates=/usr/local/bin/consoles \
80         --web.console.libraries=/usr/local/bin/console_libraries
81
82     [Install]
83     WantedBy=multi-user.target
84     when: ansible_os_family == "RedHat"
85
86     # Step 7: Reload systemd daemon
87     - name: Reload systemd daemon
88       ansible.builtin.systemd:
89         daemon_reload: yes
90
91     # Step 8: Enable and start Prometheus service
92     - name: Enable and start Prometheus service
93       ansible.builtin.systemd:
94         name: prometheus
95         state: started
96         enabled: yes
97
98     # Step 9: Update MOTD
99     - name: Get current user for MOTD
100       ansible.builtin.command:
101         cmd: whoami
102       register: current_user
103
104     - name: Update MOTD with user info
105       ansible.builtin.copy:
106         dest: /etc/motd
107         content: |
108           Ansible Managed by {{ current_user.stdout }}
109         owner: root
110         group: root
111         mode: '0644'S

```

Analysis: this is the code for the prometheus server

## Debian:

```
'all'

PLAY [Install and configure Prometheus server] *****
TASK [Gathering Facts] *****
ok: [192.168.56.112]
TASK [Install Prometheus on Debian] *****
ok: [192.168.56.112]
TASK [Install dependencies on CentOS] *****
skipping: [192.168.56.112]
TASK [Download Prometheus binary (CentOS)] *****
skipping: [192.168.56.112]
TASK [Extract Prometheus (CentOS)] *****
skipping: [192.168.56.112]
TASK [Configure Prometheus to scrape all Node Exporters] *****
changed: [192.168.56.112]
TASK [Enable and start Prometheus] *****
ok: [192.168.56.112]

PLAY RECAP *****
192.168.56.112 : ok=4  changed=1  unreachable=0  failed=0  skipped=3  rescued=0  ignored=0
```

```
● prometheus.service - Monitoring system and time series database
   Loaded: loaded (/usr/lib/systemd/system/prometheus.service; enabled; preset=
   Active: active (running) since Fri 2025-11-14 09:57:37 UTC; 19min ago
     Docs: https://prometheus.io/docs/introduction/overview/
    Main PID: 29022 (prometheus)
      Tasks: 11 (limit: 13085)
     Memory: 26.5M (peak: 30.4M)
        CPU: 1.177s
     CGroup: /system.slice/prometheus.service
             └─29022 /usr/bin/prometheus
```

Analysis: This is the Debian VM, and it is also successful and it said “active”

## CENTOS:

```
TASK [Install dependencies on CentOS (wget)] *****
ok: [192.168.56.114]

TASK [Download Prometheus binary (CentOS)] *****
ok: [192.168.56.114]

TASK [Extract Prometheus binary (CentOS)] *****
ok: [192.168.56.114]

TASK [Configure Prometheus to scrape all Node Exporters] *****
changed: [192.168.56.114]

TASK [Create Prometheus systemd service (CentOS)] *****
changed: [192.168.56.114]

TASK [Reload systemd daemon] *****
ok: [192.168.56.114]

TASK [Enable and start Prometheus service] *****
changed: [192.168.56.114]

TASK [Get current user for MOTD] *****
changed: [192.168.56.114]

TASK [Update MOTD with user info] *****
changed: [192.168.56.114]

PLAY RECAP *****
192.168.56.114      : ok=11  changed=5  unreachable=0  failed=0  skipped=1  rescued=0  ignored=0
```

```
node_exporter.service - Prometheus Node Exporter
  Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset:
  Active: active (running) since Fri 2025-11-14 17:48:06 PST; 38min ago
  Main PID: 6531 (node_exporter)
  Tasks: 6 (limit: 23007)
  Memory: 5.0M
  CPU: 132ms
  CGroup: /system.slice/node_exporter.service
          └─6531 /usr/local/bin/node_exporter

Nov 14 17:48:06 vbox node_exporter[6531]: ts=2025-11-14T09:48:06.155Z caller=no
```

Analysis: It says active so the tasks and installation was successful in CentOS VM

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

Code:

```
65
66     # 4.4 MOTD Message
67     - name: Update MOTD
68       ansible.builtin.copy:
69         dest: /etc/motd
70         content: "Ansible Managed by {{ motd_user | default('unknown_user') }}\n"
71         owner: root
72         group: root
73         mode: '0644'
74
```

```
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro
```

```
Expanded Security Maintenance for Infrastructure is not enabled.
```

```
0 updates can be applied immediately.
```

```
263 additional security updates can be applied with ESM Infra.
Learn more about enabling ESM Infra service for Ubuntu 18.04 at
https://ubuntu.com/18-04
```















```
New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```



```
Your Hardware Enablement Stack (HWE) is supported until April 2023.
Ansible Managed by lim
Last login: Thu Nov 20 05:43:16 2025 from 192.168.229.128
```

```
TASK [Update MOTD] *****
changed: [192.168.56.114]
```

Analysis: I successfully change the Motd as the “ansible managed” by my username. Based on the output the task updated the MOTD successfully.

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

 <b>joshuabueno123</b> Add files via upload		bd0dee1 · 31 minutes ago	 7 Commits
 MOTD.yml	Add files via upload	31 minutes ago	
 README.md	Initial commit	last week	
 ansible.cfg	Add files via upload	last week	
 config.yaml	Add files via upload	last week	
 enterprise.yml	Add files via upload	last week	
 enterpriseCENTOS.yml	Add files via upload	last week	
 inventory.ini	Add files via upload	last week	
 inventory.yaml	Add files via upload	last week	
 node_exporterCENTOS.yaml	Add files via upload	31 minutes ago	
 nodeexport_debian.yml	Add files via upload	last week	
 prometheus.yml	Add files via upload	31 minutes ago	
 site.yaml	Add files via upload	31 minutes ago	

 **README** 

## FINAL\_EXAM\_Bueno

Analysis: I successfully push and commit all the files that I used in the exam in my github account

5. Make sure to show evidence of input (codes) process (codes successfully running) and output (evidence of installation)

Code:

```

! prometheus.yml
1 ---
2 - name: Install and configure Prometheus server
3   hosts: prometheus_server2
4   become: true
5   vars_files:
6   - config.yml
7
8   tasks:
9     # Ensure the /etc/prometheus directory exists
10    - name: Ensure the /etc/prometheus directory exists
11      ansible.builtin.file:
12        path: /etc/prometheus
13        state: directory
14        mode: '0755'
15
16    # Step 1: Install Prometheus on Debian (APT)
17    - name: Install Prometheus on Debian
18      ansible.builtin.apt:
19        update_cache: yes
20        name: prometheus
21        state: present
22      when: ansible_os_family == "Debian"
23
24    # Step 2: Install dependencies on CentOS
25    - name: Install dependencies on CentOS (wget)
26      ansible.builtin.yum:
27        name: wget
28        state: present
29      when: ansible_os_family == "RedHat"
30
31    # Step 3: Download Prometheus binary (CentOS)
32    - name: Download Prometheus binary (CentOS)
33      ansible.builtin.get_url:
34        url: "https://github.com/prometheus/prometheus/releases/download/v2.53.1/p
35        dest: /tmp/prometheus.tar.gz

```

! prometheus.yml

```
35     dest: /tmp/prometheus.tar.gz
36     when: ansible_os_family == "RedHat"
37
38     # Step 4: Extract Prometheus binary (CentOS)
39     - name: Extract Prometheus binary (CentOS)
40       ansible.builtin.unarchive:
41         src: /tmp/prometheus.tar.gz
42         dest: /usr/local/bin/
43         extra_opts: ["--strip-components=1"]
44         remote_src: yes
45       when: ansible_os_family == "RedHat"
46
47     # Step 5: Configure Prometheus to scrape all Node Exporters
48     - name: Configure Prometheus to scrape all Node Exporters
49       ansible.builtin.copy:
50         dest: /etc/prometheus/prometheus.yml
51         content: |
52             global:
53               scrape_interval: 15s
54             scrape_configs:
55               - job_name: "node_exporter"
56                 static_configs:
57                   - targets:
58                       {% for host in groups['all_node_exporter'] %}
59                       - "{{ host }}:9100"
60                       {% endfor %}
61
62     # Step 6: Create Prometheus systemd service (for CentOS)
63     - name: Create Prometheus systemd service (CentOS)
64       ansible.builtin.copy:
65         dest: /etc/systemd/system/prometheus.service
66         content: |
67             [Unit]
68             Description=Prometheus Monitoring
```



```
! prometheus.yml
69     Wants=network-online.target
70     After=network-online.target
71
72     [Service]
73     User=nobody
74     Group=nogroup
75     Type=simple
76     ExecStart=/usr/local/bin/prometheus \
77         --config.file /etc/prometheus/prometheus.yml \
78         --storage.tsdb.path /var/lib/prometheus \
79         --web.console.templates=/usr/local/bin/consoles \
80         --web.console.libraries=/usr/local/bin/console_libraries
81
82     [Install]
83     WantedBy=multi-user.target
84     when: ansible_os_family == "RedHat"
85
86     # Step 7: Reload systemd daemon
87     - name: Reload systemd daemon
88       ansible.builtin.systemd:
89         daemon_reload: yes
90
91     # Step 8: Enable and start Prometheus service
92     - name: Enable and start Prometheus service
93       ansible.builtin.systemd:
94         name: prometheus
95         state: started
96         enabled: yes
97
98     # Step 9: Update MOTD
99     - name: Get current user for MOTD
100       ansible.builtin.command:
101         cmd: whoami
102       register: current_user
```

```

! prometheus.yml
69     Wants=network-online.target
70     After=network-online.target
71
72     [Service]
73     User=nobody
74     Group=nogroup
75     Type=simple
76     ExecStart=/usr/local/bin/prometheus \
77         --config.file /etc/prometheus/prometheus.yml \
78         --storage.tsdb.path /var/lib/prometheus \
79         --web.console.templates=/usr/local/bin/consoles \
80         --web.console.libraries=/usr/local/bin/console_libraries
81
82     [Install]
83     WantedBy=multi-user.target
84     when: ansible_os_family == "RedHat"
85
86     # Step 7: Reload systemd daemon
87     - name: Reload systemd daemon
88       ansible.builtin.systemd:
89         daemon_reload: yes
90
91     # Step 8: Enable and start Prometheus service
92     - name: Enable and start Prometheus service
93       ansible.builtin.systemd:
94         name: prometheus
95         state: started
96         enabled: yes
97
98     # Step 9: Update MOTD
99     - name: Get current user for MOTD
100       ansible.builtin.command:
101         cmd: whoami
102       register: current_user
103
104     - name: Update MOTD with user info
105       ansible.builtin.copy:
106         dest: /etc/motd
107         content: |
108             Ansible Managed by {{ current_user.stdout }}
109         owner: root
110         group: root
111         mode: '0644'S

```

Analysis: This is the code in the installation of the tasks, the prometheus, node exporter, motd.

Process:

```

'all'

PLAY [Install and configure Prometheus server] *****
TASK [Gathering Facts] *****
ok: [192.168.56.112]

TASK [Install Prometheus on Debian] *****
ok: [192.168.56.112]

TASK [Install dependencies on CentOS] *****
skipping: [192.168.56.112]

TASK [Download Prometheus binary (CentOS)] *****
skipping: [192.168.56.112]

TASK [Extract Prometheus (CentOS)] *****
skipping: [192.168.56.112]

TASK [Configure Prometheus to scrape all Node Exporters] *****
changed: [192.168.56.112]

TASK [Enable and start Prometheus] *****
ok: [192.168.56.112]

PLAY RECAP *****
192.168.56.112      : ok=4   changed=1   unreachable=0   failed=0   skipped=3   rescued=0   ignored=0

```

Analysis: This is the process of installing all the required tasks in Debian  
Output:

```

TASK [Install dependencies on CentOS (wget)] *****
ok: [192.168.56.114]

TASK [Download Prometheus binary (CentOS)] *****
ok: [192.168.56.114]

TASK [Extract Prometheus binary (CentOS)] *****
ok: [192.168.56.114]

TASK [Configure Prometheus to scrape all Node Exporters] *****
changed: [192.168.56.114]

TASK [Create Prometheus systemd service (CentOS)] *****
changed: [192.168.56.114]

TASK [Reload systemd daemon] *****
ok: [192.168.56.114]

TASK [Enable and start Prometheus service] *****
changed: [192.168.56.114]

TASK [Get current user for MOTD] *****
changed: [192.168.56.114]

TASK [Update MOTD with user info] *****
changed: [192.168.56.114]

PLAY RECAP *****
192.168.56.114      : ok=11  changed=5   unreachable=0   failed=0   skipped=1   rescued=0   ignored=0

```

Analysis: As you can see in the output, all the installation was successful

```
node_exporter.service - Prometheus Node Exporter
  Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset
  Active: active (running) since Fri 2025-11-14 17:48:06 PST; 38min ago
  Main PID: 6531 (node_exporter)
  Tasks: 6 (limit: 23007)
  Memory: 5.0M
  CPU: 132ms
  CGroup: /system.slice/node_exporter.service
          └─6531 /usr/local/bin/node_exporter
```

Analysis: It says active also, the installation was successful also in CentOS

5. For your final exam to be counted, please paste your repository link as an answer in this exam.

**Github Link:**

[https://github.com/joshuabueno123/FINAL\\_EXAM\\_Bueno.git](https://github.com/joshuabueno123/FINAL_EXAM_Bueno.git)

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