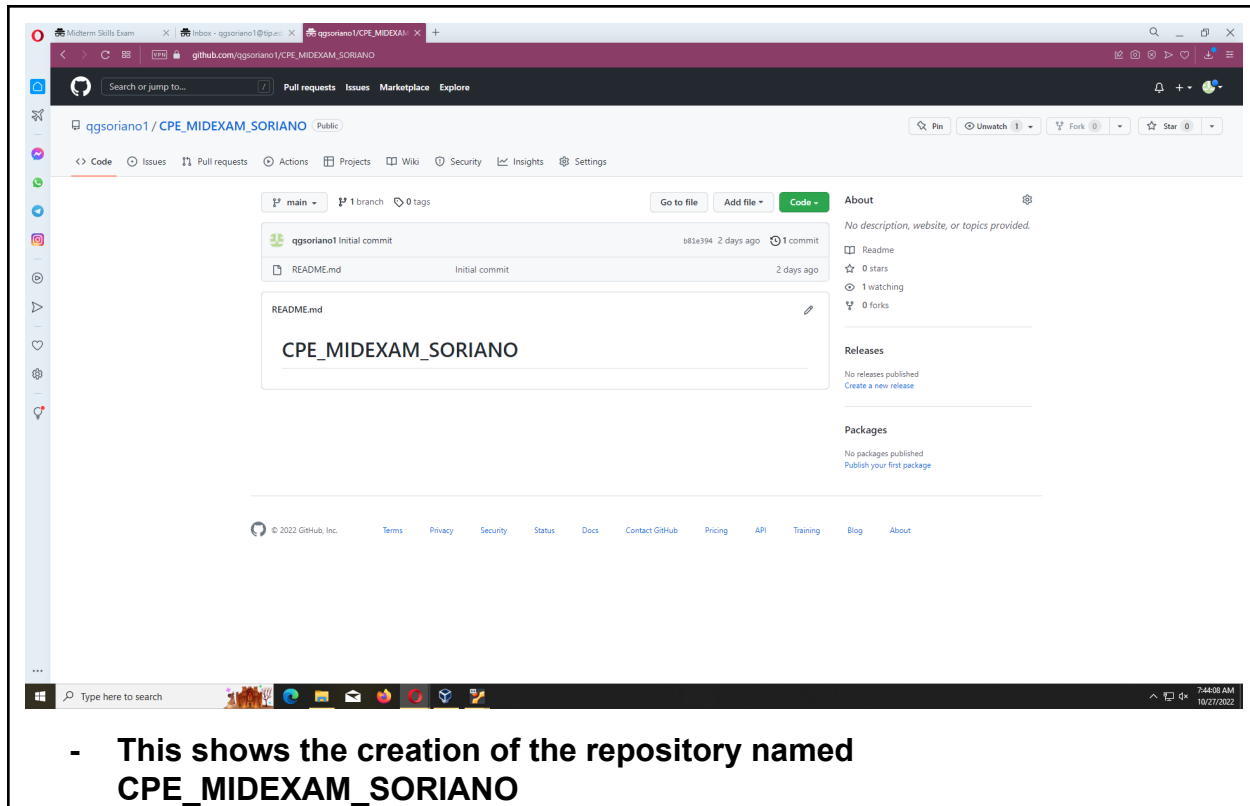
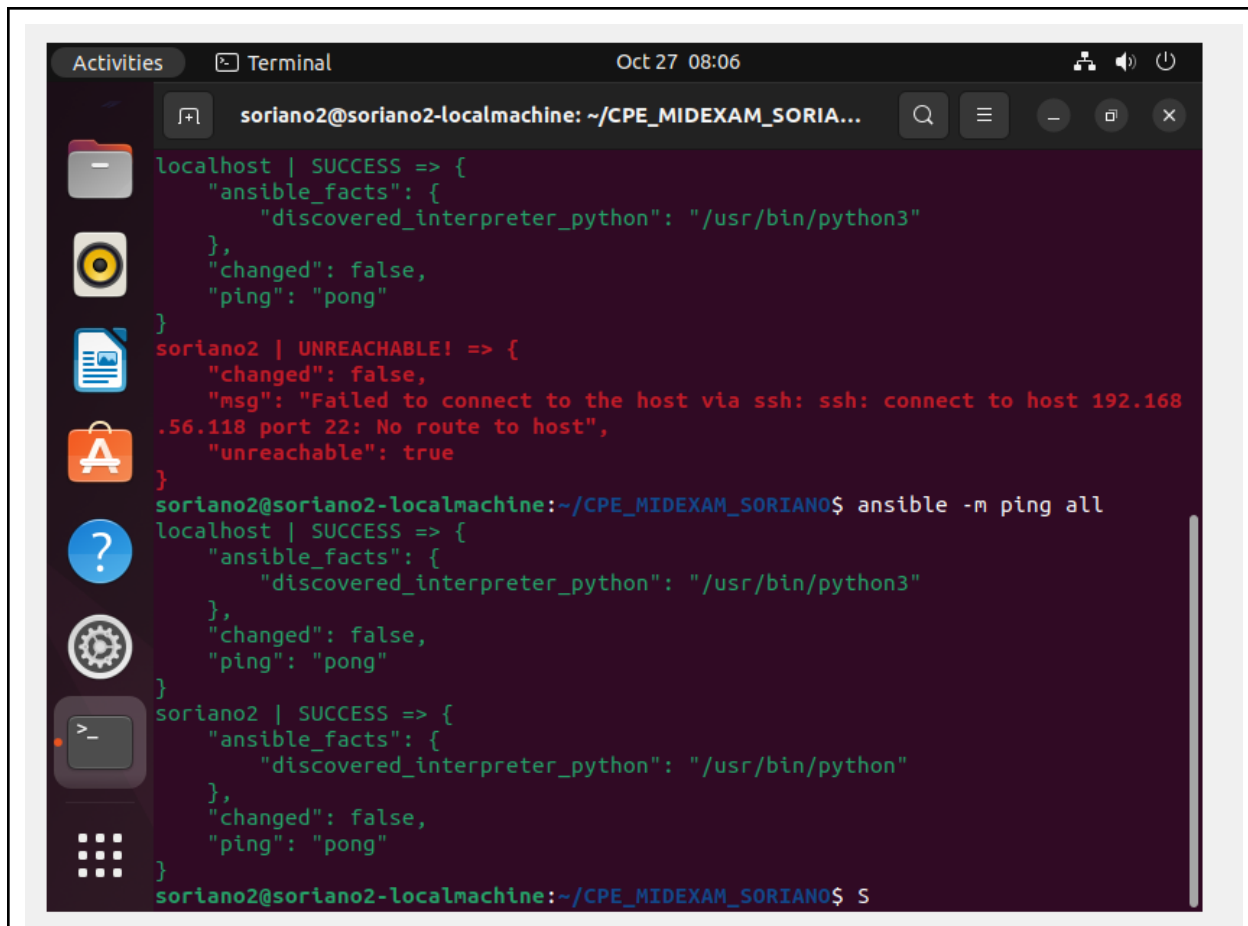


Name: Gabriel Soriano	Date Performed: October 27, 2022
Course/Section: CPE 232 - CPE31S23	Date Submitted: October 27, 2022
Instructor: Engr. Taylar	Semester and SY: 1st sem- SY 2022-2023
Midterm Skills Exam: Install, Configure, and Manage Log Monitoring tools	
1. Objectives	
Create and design a workflow that installs, configure and manage enterprise availability, performance and log monitoring tools using Ansible as an Infrastructure as Code (IaC) tool.	
2. Instructions	
<ol style="list-style-type: none"> 1. Create a repository in your GitHub account and label it CPE_MIDEXAM_SURNAME. DONE 2. Clone the repository and do the following: <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: DONE 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) DONE 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)	
!1 SCREENSHOTS/PROOFS:	
SCREENSHOTS:	



- This shows the creation of the repository named **CPE_MIDEXAM_SORIANO**

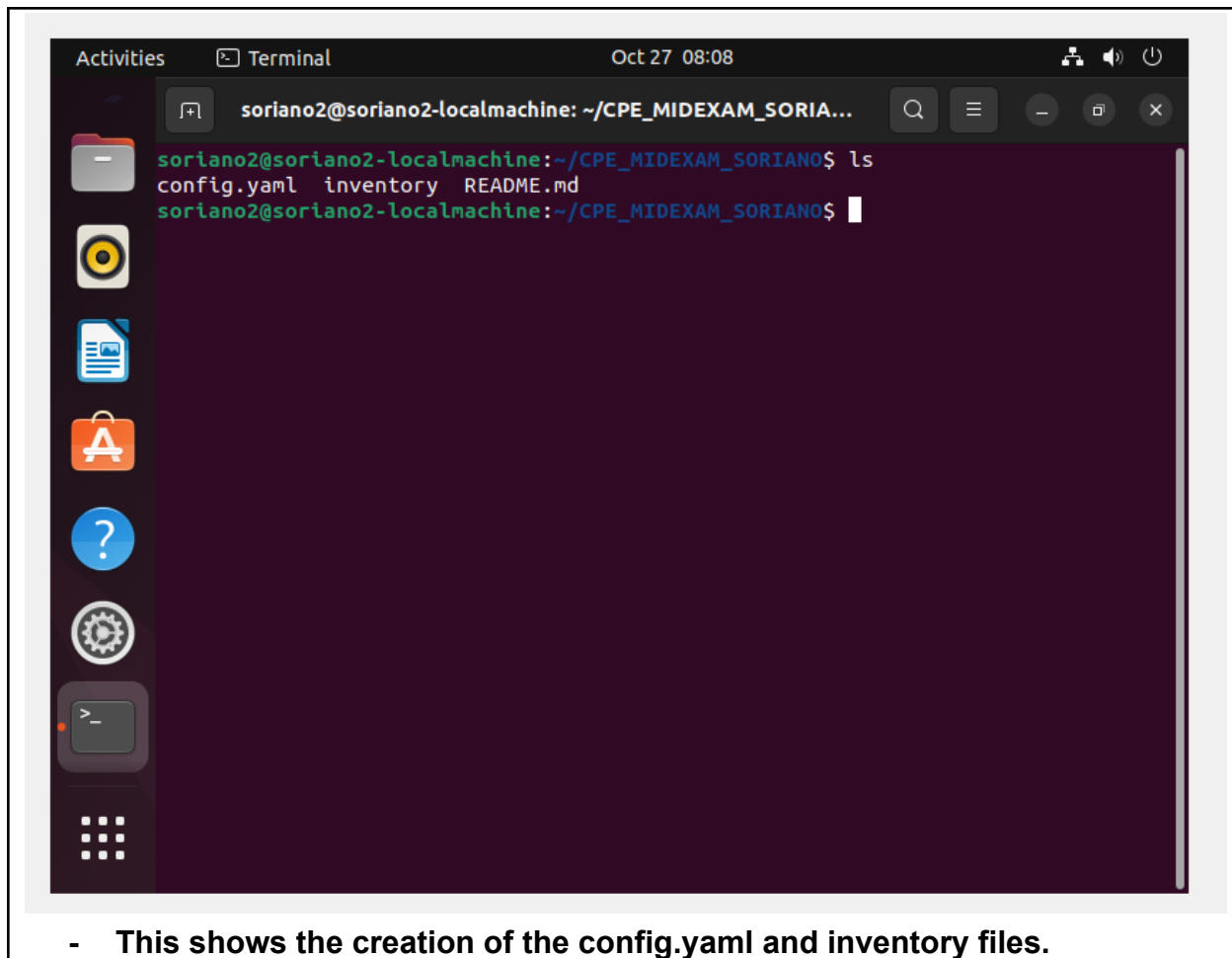
A screenshot of a Linux terminal window. The title bar shows 'Activities', 'Terminal', and the date 'Oct 27 08:06'. The terminal prompt is 'soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...'. The output shows the results of an Ansible command: 'localhost | SUCCESS => {', 'soriano2 | UNREACHABLE! => {', and 'soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO\$ ansible -m ping all'. The output for 'localhost' and 'soriano2' shows 'SUCCESS' with 'ping: pong'. The output for 'soriano2' also shows 'UNREACHABLE!'. The terminal window has a sidebar with icons for file manager, applications, and settings.

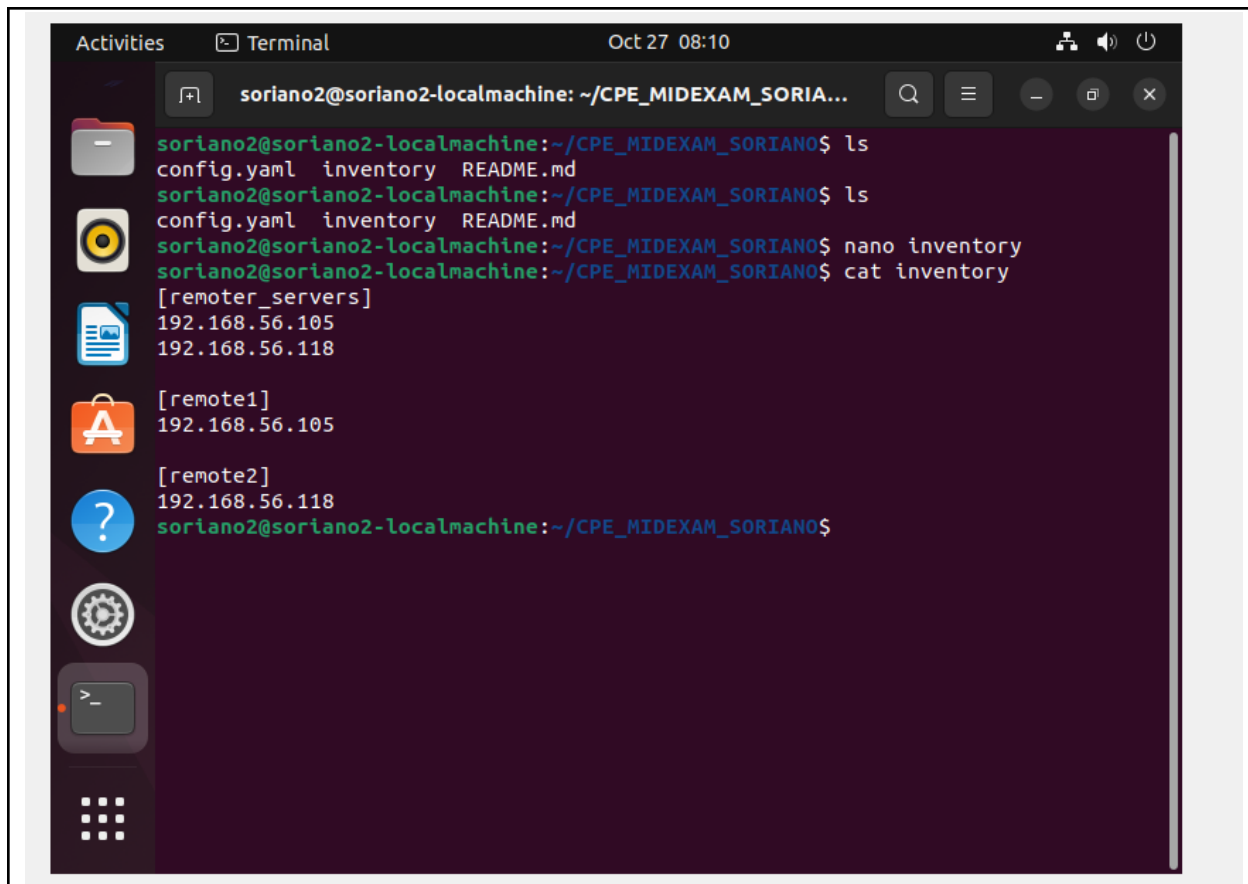
```
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
soriano2 | UNREACHABLE! => {
  "changed": false,
  "msg": "Failed to connect to the host via ssh: ssh: connect to host 192.168.56.118 port 22: No route to host",
  "unreachable": true
}
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$ ansible -m ping all
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
soriano2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$ s
```

- This is the proof that all of the remote servers are reachable without errors. This is known with the command `ansible -m ping all`.

!2.1 SCREENSHOTS/PROOFS:

SCREENSHOT:

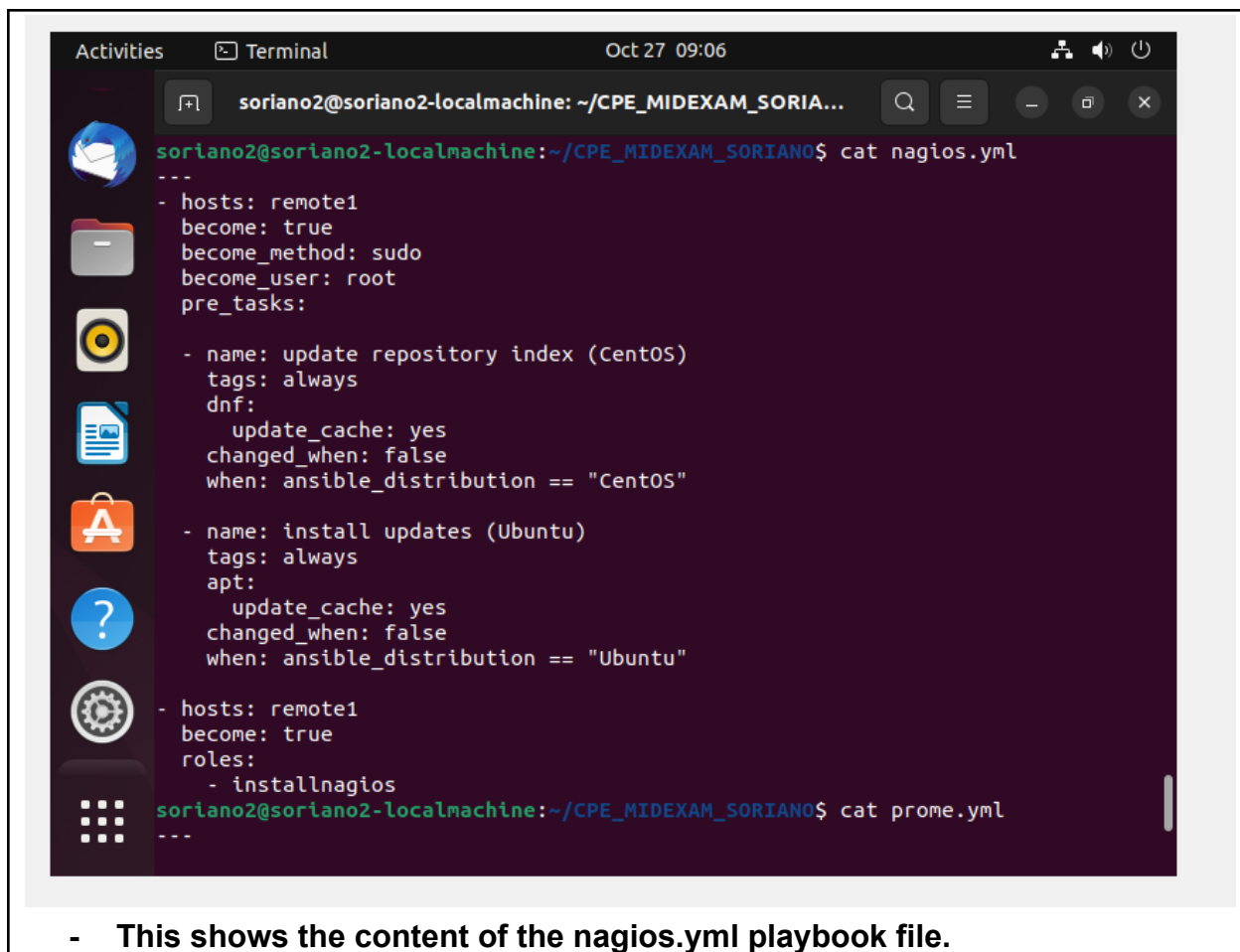


A screenshot of a Linux terminal window. The window title is "Terminal" and the date/time is "Oct 27 08:10". The terminal shows the user "soriano2" at "soriano2-localmachine" in the directory "~/CPE_MIDEXAM_SORIANO". The user runs "ls" and lists "config.yaml", "inventory", and "README.md". Then, they run "nano inventory" and "cat inventory", which displays the content of the inventory file. The content shows a group named "remoter_servers" with two hosts: "192.168.56.105" (labeled "remote1") and "192.168.56.118" (labeled "remote2"). The terminal window has a sidebar with various application icons and standard window controls.

- This shows the content of the arranged inventory file. The purpose of this is to make a playbook installation only installed on a specific remote server only. Some of the requirements only need to be installed on only one host/workstation.

!2.2 SCREENSHOTS/PROOFS:

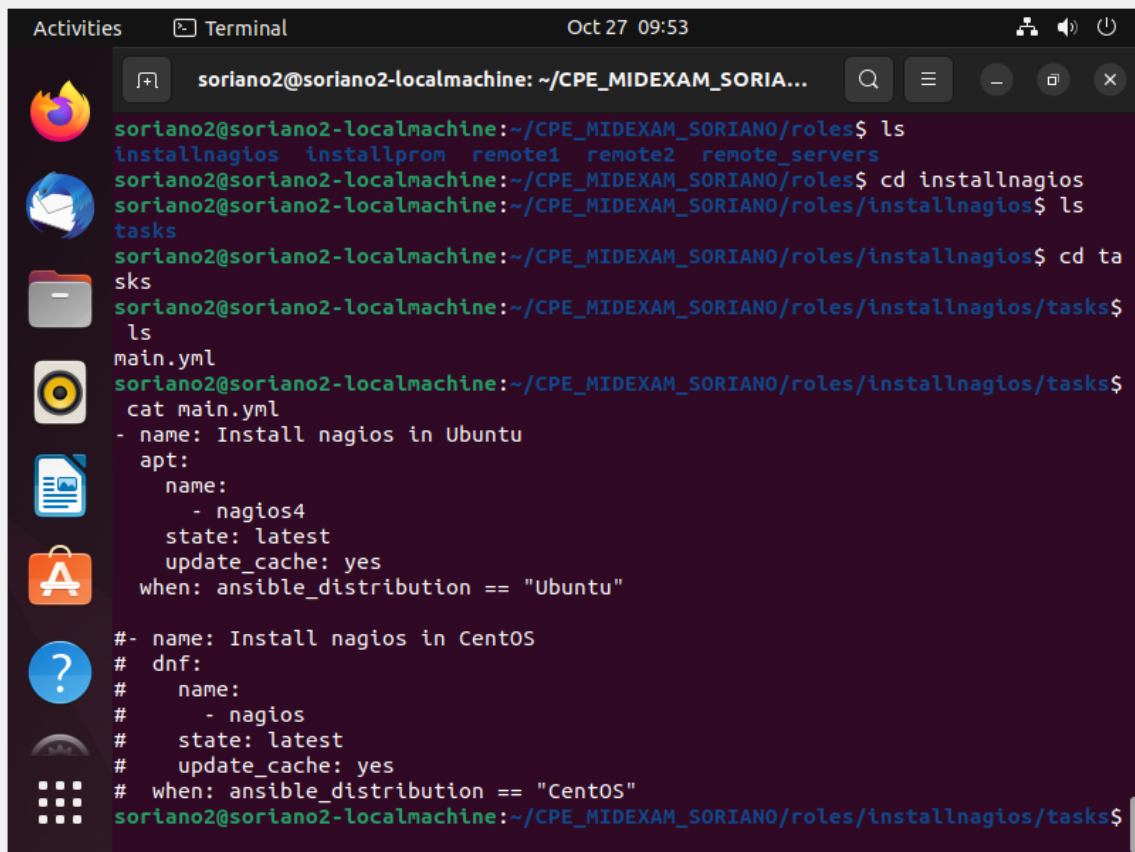
SCREENSHOTS:



The image shows a terminal window titled "Terminal" with the date and time "Oct 27 09:06". The terminal is running a command to display the contents of a file named "nagios.yml". The output shows a list of tasks for an Ansible playbook. The tasks are organized into two main sections, each starting with a hyphen and "hosts: remote1". The first section includes tasks for updating the repository index on CentOS and installing updates on Ubuntu. The second section includes a task for installing Nagios on remote1. The terminal prompt is "soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIANO\$".

```
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIANO$ cat nagios.yml
---
- hosts: remote1
  become: true
  become_method: sudo
  become_user: root
  pre_tasks:
    - name: update repository index (CentOS)
      tags: always
      dnf:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "CentOS"
    - name: install updates (Ubuntu)
      tags: always
      apt:
        update_cache: yes
        changed_when: false
        when: ansible_distribution == "Ubuntu"
- hosts: remote1
  become: true
  roles:
    - installnagios
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIANO$ cat prome.yml
---
```

- This shows the content of the nagios.yml playbook file.



```
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIANO/roles$ ls
installnagios  installprom  remote1  remote2  remote_servers
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles$ cd installnagios
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installnagios$ ls
tasks
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installnagios$ cd tasks
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installnagios/tasks$ ls
main.yml
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installnagios/tasks$ cat main.yml
- name: Install nagios in Ubuntu
  apt:
    name:
      - nagios4
    state: latest
    update_cache: yes
    when: ansible_distribution == "Ubuntu"

#- name: Install nagios in CentOS
#  dnf:
#    name:
#      - nagios
#    state: latest
#    update_cache: yes
#  when: ansible_distribution == "CentOS"
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installnagios/tasks$
```

- This shows the content of the main.yml playbook file for the installation of the nagios on only one workstation, which is the ubuntu workstation only.

Activities Terminal Oct 27 09:03

soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...

ansible.cfg elstack.yml infdb.yml lampstack.yml prome.yml roles
config.yaml graf.yml inventory nagios.yml README.md

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO\$ ansible-playbook --ask-be
come-pass nagios.yml
BECOME password:

PLAY [remote1] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.105]

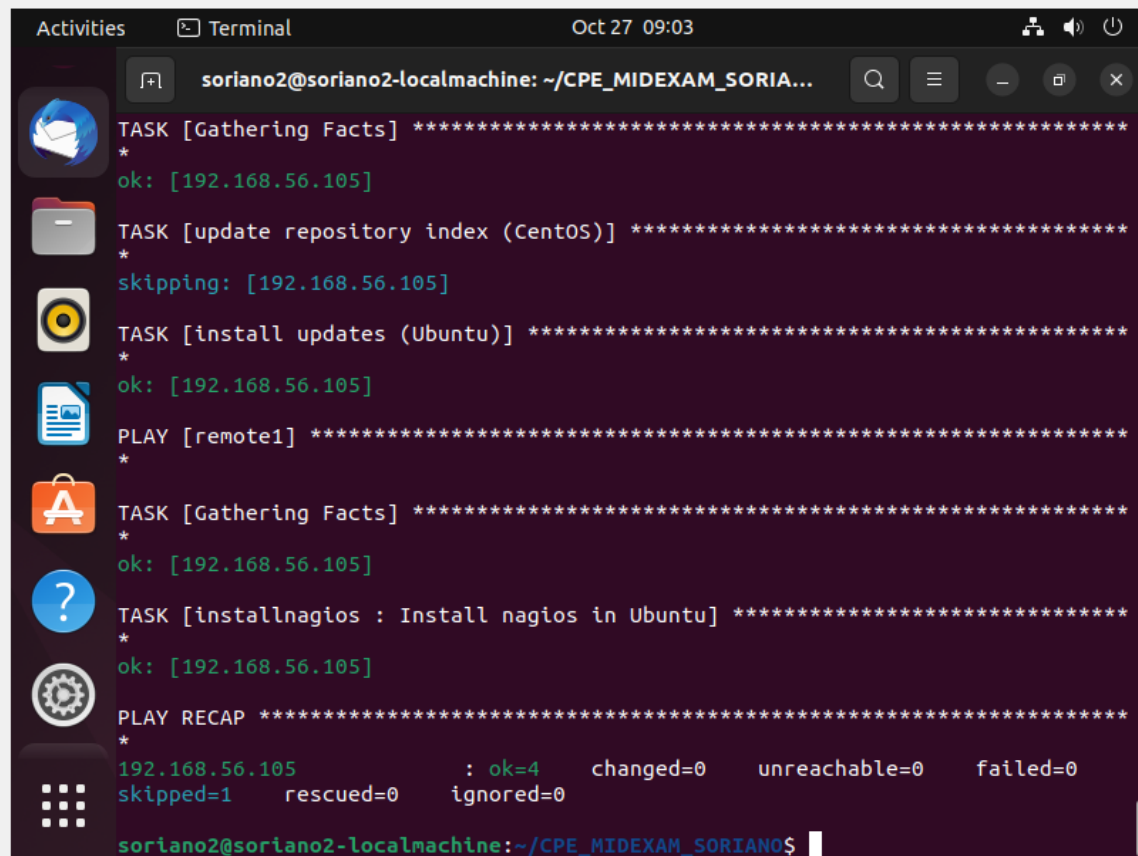
TASK [update repository index (CentOS)] *****
*
skipping: [192.168.56.105]

TASK [install updates (Ubuntu)] *****
*
ok: [192.168.56.105]

PLAY [remote1] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.105]

TASK [installnagios : Install nagios in Ubuntu] *****



```
Activities Terminal Oct 27 09:03
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...

TASK [Gathering Facts] *****
*
ok: [192.168.56.105]

TASK [update repository index (CentOS)] *****
*
skipping: [192.168.56.105]

TASK [install updates (Ubuntu)] *****
*
ok: [192.168.56.105]

PLAY [remote1] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.105]










TASK [installnagios : Install nagios in Ubuntu] *****
*
ok: [192.168.56.105]

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

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

- This shows the successful run of the playbook file named nagio.yml. As seen above, there are no errors and the run process is successful.

ActivitiesTerminalOct 27 09:03soriano2@server1-Server1: ~



```
soriano2@server1-Server1:~$ nagios4 --version

Nagios Core 4.4.6
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2020-04-28
License: GPL

Website: https://www.nagios.org

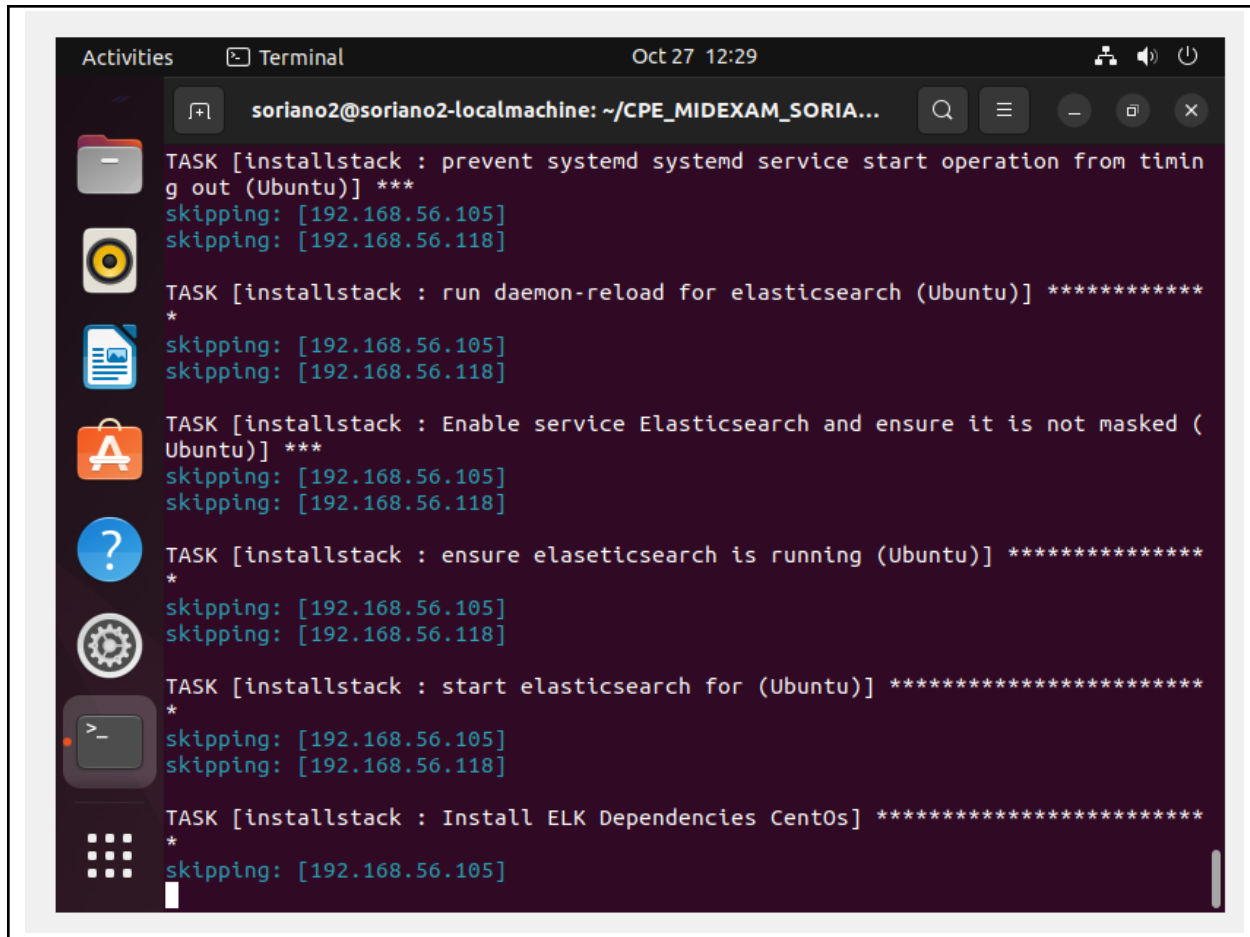
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 2 as published by the Free Software Foundation.

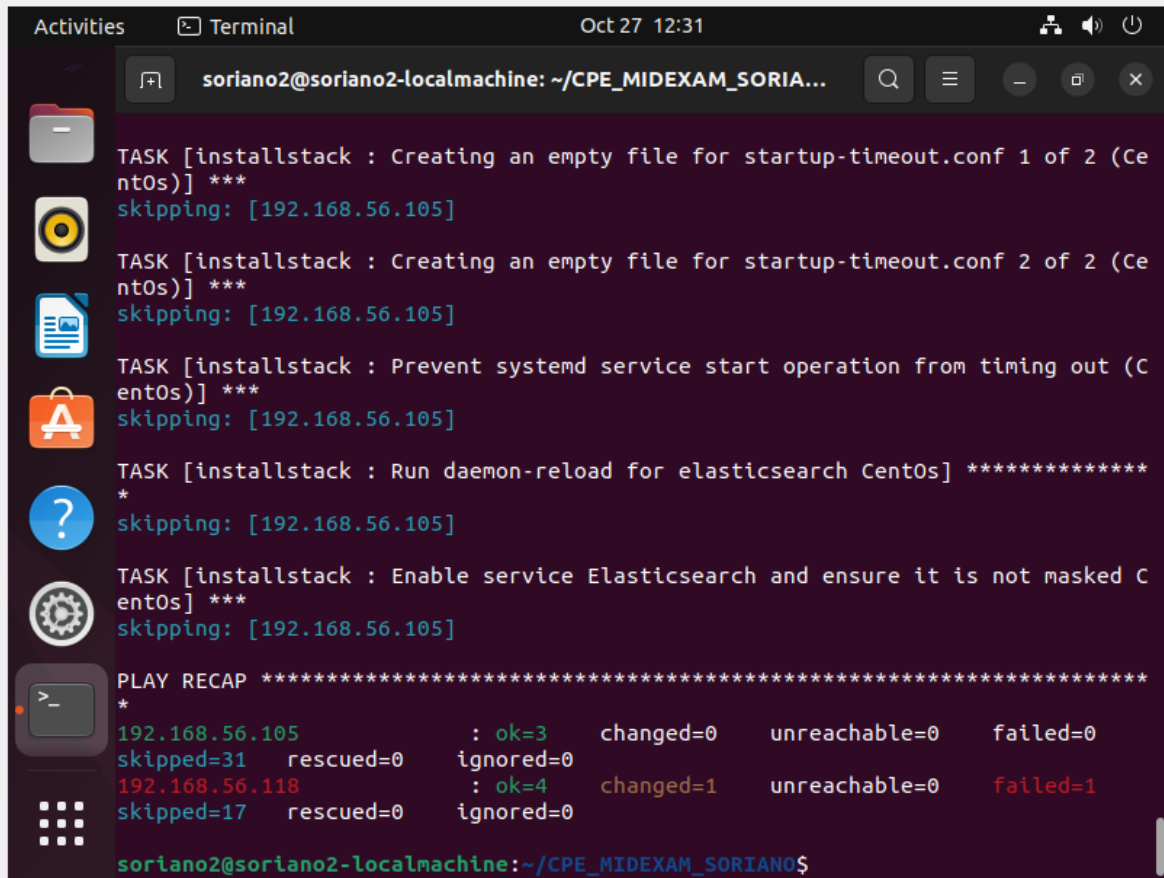
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

soriano2@server1-Server1:~$
```

- This is the proof that the nagios has been successfully installed at only one server, which is the ubuntu host/workstation.





```
Activities Terminal Oct 27 12:31
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...

TASK [installstack : Creating an empty file for startup-timeout.conf 1 of 2 (CentOs)] ***
skipping: [192.168.56.105]

TASK [installstack : Creating an empty file for startup-timeout.conf 2 of 2 (CentOs)] ***
skipping: [192.168.56.105]

TASK [installstack : Prevent systemd service start operation from timing out (CentOs)] ***
skipping: [192.168.56.105]

TASK [installstack : Run daemon-reload for elasticsearch CentOs] *****
*
skipping: [192.168.56.105]

TASK [installstack : Enable service Elasticsearch and ensure it is not masked CentOs] ***
skipping: [192.168.56.105]

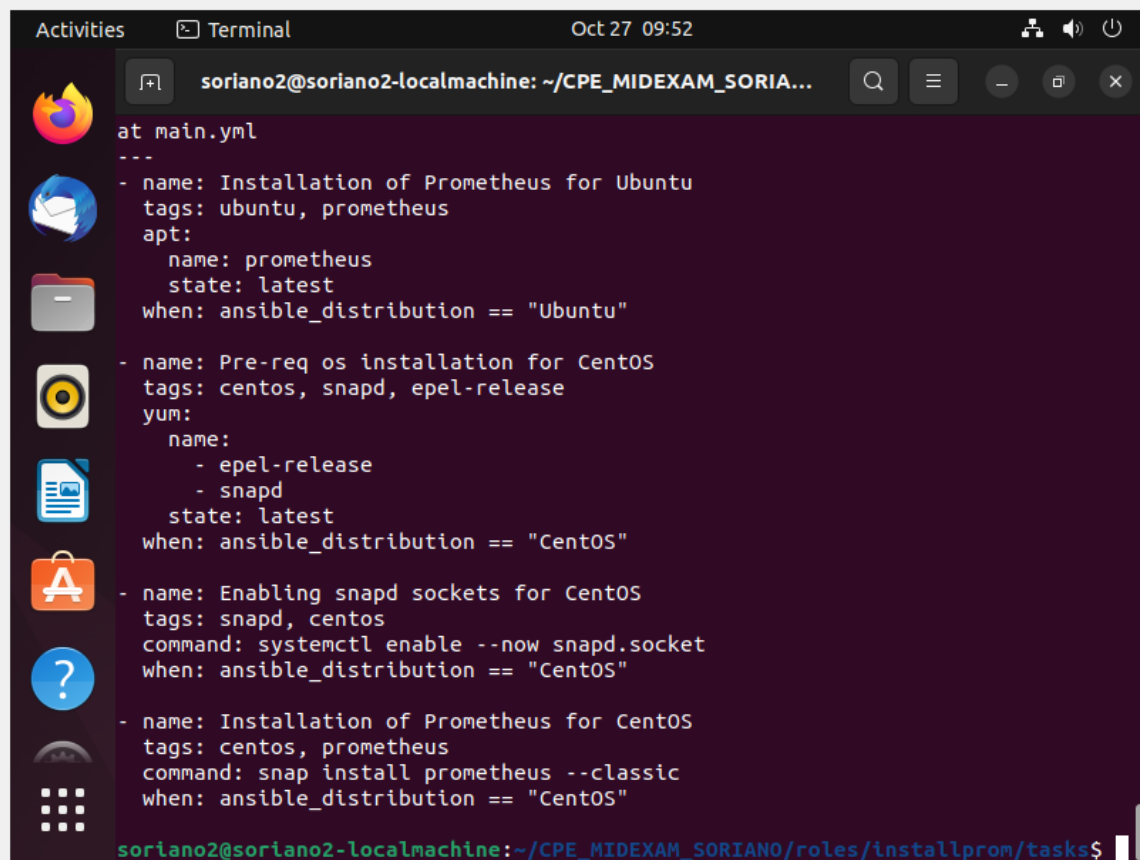
PLAY RECAP *****
*
192.168.56.105 : ok=3 changed=0 unreachable=0 failed=0
skipped=31 rescued=0 ignored=0
192.168.56.118 : ok=4 changed=1 unreachable=0 failed=1
skipped=17 rescued=0 ignored=0

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

- This shows the run of the playbook file for the installation of the elastack.

12.3 SCREENSHOTS/PROOFS:

SCREENSHOTS:



The image shows a terminal window titled "Terminal" with the date and time "Oct 27 09:52". The user is logged in as "soriano2" on a "soriano2-localmachine". The terminal displays the content of a file named "main.yml", which is an Ansible playbook for installing Prometheus. The playbook has two main sections: one for Ubuntu and one for CentOS. The Ubuntu section includes a task to install Prometheus using the 'apt' package manager. The CentOS section includes tasks to install prerequisites (epel-release and snapd) using 'yum', enable snapd sockets using 'systemctl', and finally install Prometheus using 'snap'.

```
at main.yml
---
- name: Installation of Prometheus for Ubuntu
  tags: ubuntu, prometheus
  apt:
    name: prometheus
    state: latest
    when: ansible_distribution == "Ubuntu"

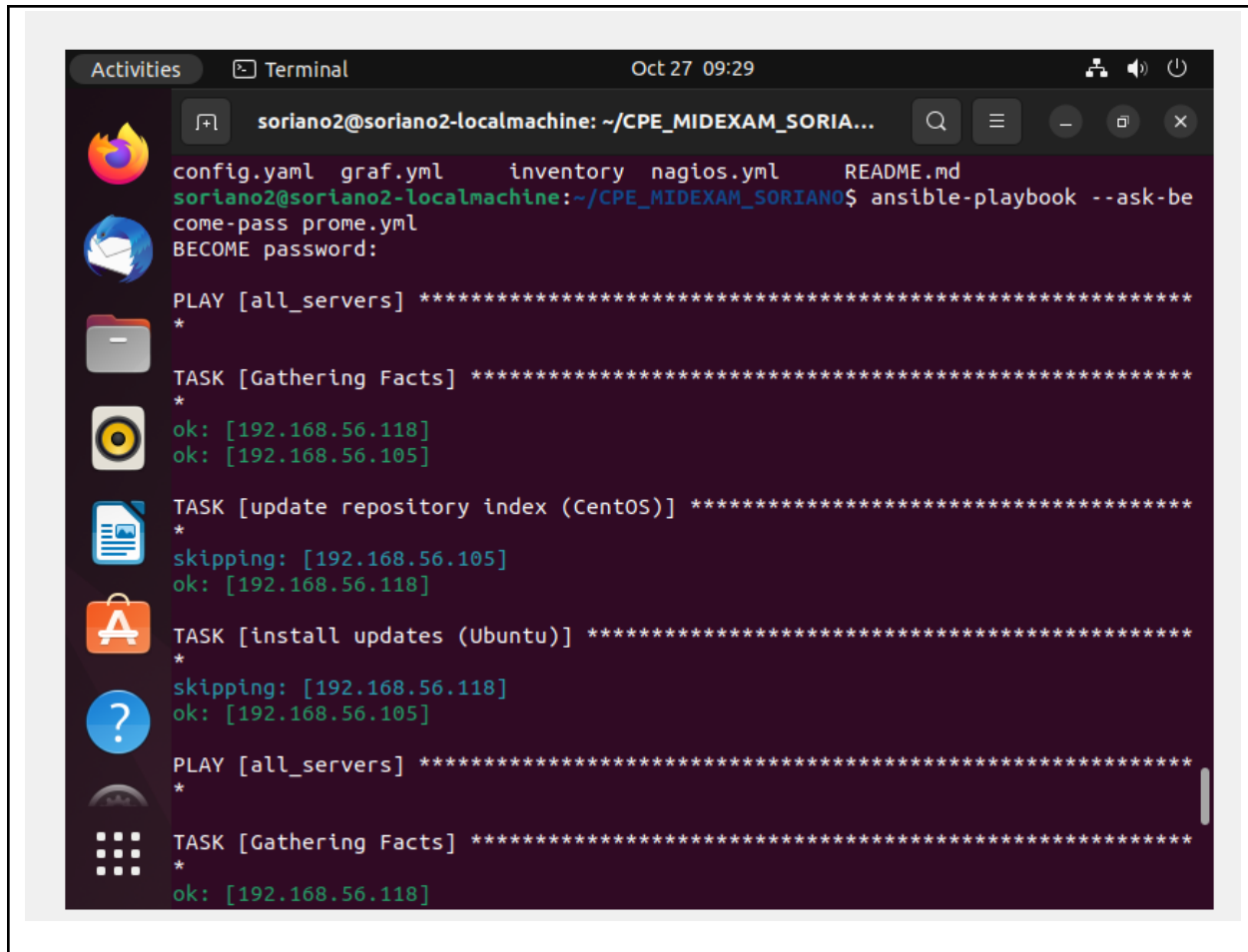
- name: Pre-req os installation for CentOS
  tags: centos, snapd, epel-release
  yum:
    name:
      - epel-release
      - snapd
    state: latest
    when: ansible_distribution == "CentOS"

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

- name: Installation of Prometheus for CentOS
  tags: centos, prometheus
  command: snap install prometheus --classic
  when: ansible_distribution == "CentOS"

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO/roles/installprom/tasks$
```

- This shows the content of the main.yml playbook file for the installation of the prometheus on both ubuntu and centOS workstations.





soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...



TASK [Gathering Facts] *****

ok: [192.168.56.118]

ok: [192.168.56.105]



TASK [installprom : Installation of Prometheus for Ubuntu] *****

skipping: [192.168.56.118]

changed: [192.168.56.105]



TASK [installprom : Pre-req os installation for CentOS] *****

skipping: [192.168.56.105]

changed: [192.168.56.118]



TASK [installprom : Enabling snapd sockets for CentOS] *****

skipping: [192.168.56.105]

changed: [192.168.56.118]



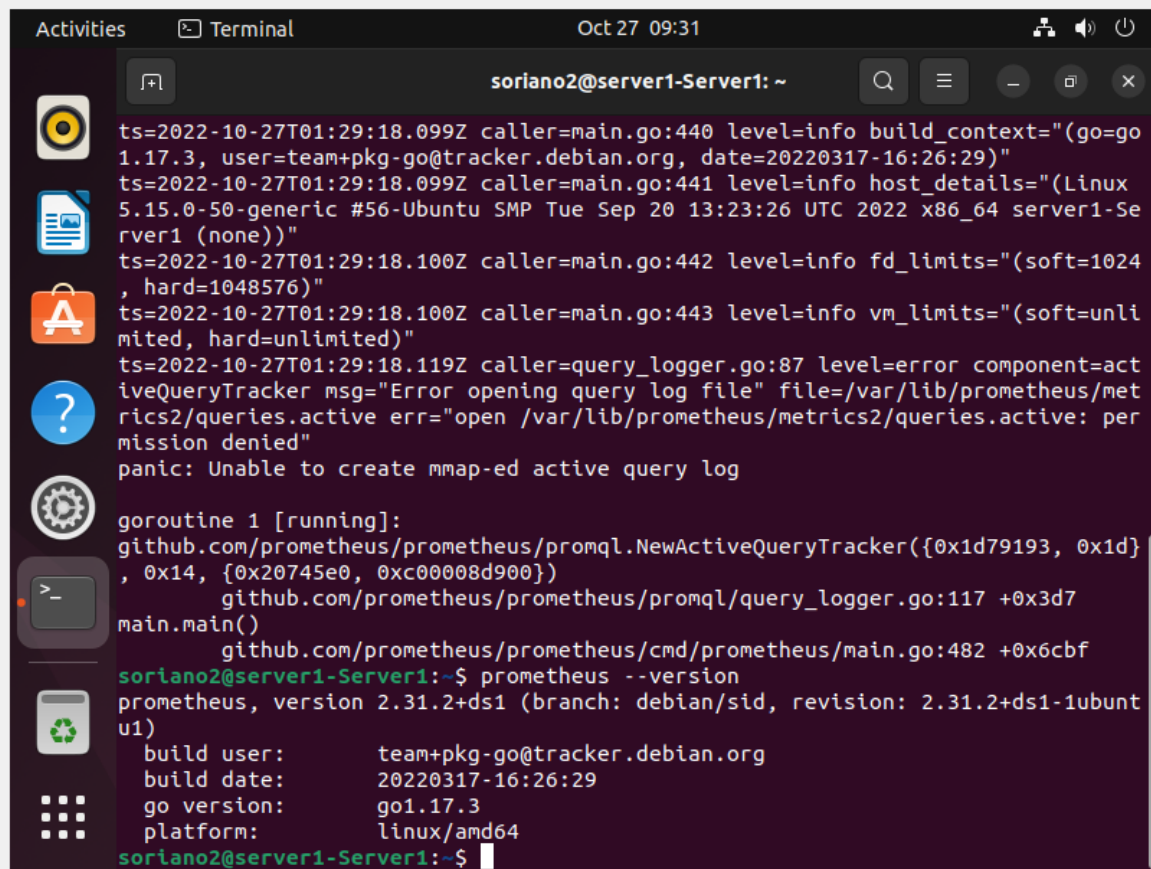
TASK [installprom : Installation of Prometheus for CentOS] *****

skipping: [192.168.56.105]

fatal: [192.168.56.118]: FAILED! => {"changed": true, "cmd": ["snap", "install", "prometheus", "--classic"], "delta": "0:00:00.594823", "end": "2022-10-26 21:28:27.092936", "msg": "non-zero return code", "rc": 10, "start": "2022-10-26 21:28:26.498113", "stderr": "error: too early for operation, device not yet seede



ActivitiesTerminalOct 27 09:30soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...TASK [installprom : Pre-req os installation for CentOS] *****skipping: [192.168.56.105]changed: [192.168.56.118]TASK [installprom : Enabling snapd sockets for CentOS] *****skipping: [192.168.56.105]changed: [192.168.56.118]TASK [installprom : Installation of Prometheus for CentOS] *****fatal: [192.168.56.118]: FAILED! => {"changed": true, "cmd": ["snap", "install", "prometheus", "--classic"], "delta": "0:00:00.594823", "end": "2022-10-26 21:28:27.092936", "msg": "non-zero return code", "rc": 10, "start": "2022-10-26 21:28:26.498113", "stderr": "error: too early for operation, device not yet seeded or device model not acknowledged", "stderr_lines": ["error: too early for operation, device not yet seeded or device model not acknowledged"], "stdout": "", "stdout_lines": []}PLAY RECAP *****192.168.56.105: ok=4 changed=1 unreachable=0 failed=0skipped=4 rescued=0 ignored=0192.168.56.118: ok=5 changed=2 unreachable=0 failed=1skipped=2 rescued=0 ignored=0soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO\$

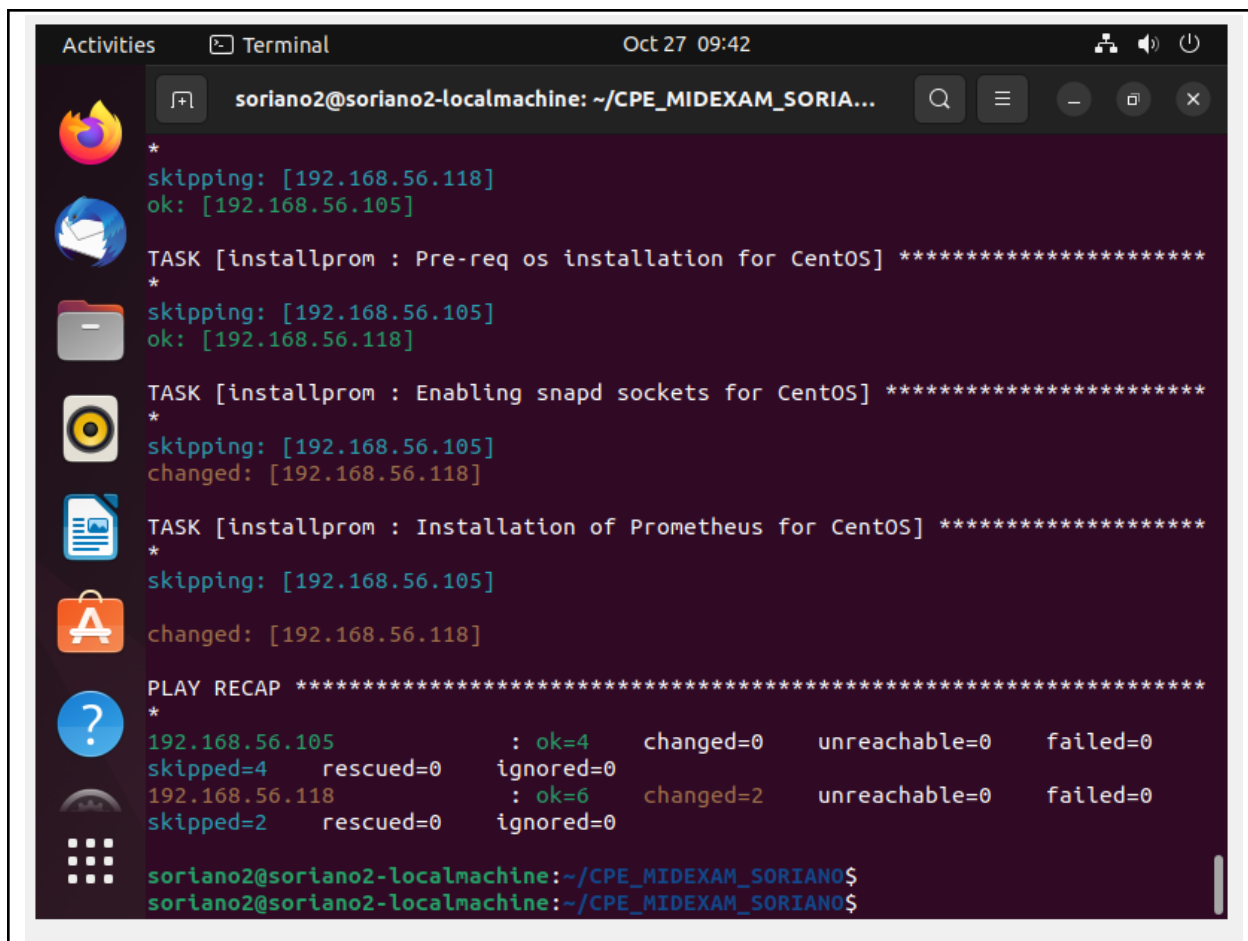


The image shows a terminal window titled 'Terminal' with the date 'Oct 27 09:31'. The prompt is 'soriano2@server1-Server1: ~'. The terminal output consists of several log lines with timestamps and caller information, followed by a stack trace for a panic, and finally the output of the 'prometheus --version' command.

```
ts=2022-10-27T01:29:18.099Z caller=main.go:440 level=info build_context="(go=go
1.17.3, user=team+pkg-go@tracker.debian.org, date=20220317-16:26:29)"
ts=2022-10-27T01:29:18.099Z caller=main.go:441 level=info host_details="(Linux
5.15.0-50-generic #56-Ubuntu SMP Tue Sep 20 13:23:26 UTC 2022 x86_64 server1-Se
rver1 (none))"
ts=2022-10-27T01:29:18.100Z caller=main.go:442 level=info fd_limits="(soft=1024
, hard=1048576)"
ts=2022-10-27T01:29:18.100Z caller=main.go:443 level=info vm_limits="(soft=unli
mited, hard=unlimited)"
ts=2022-10-27T01:29:18.119Z caller=query_logger.go:87 level=error component=act
iveQueryTracker msg="Error opening query log file" file=/var/lib/prometheus/met
rics2/queries.active err="open /var/lib/prometheus/metrics2/queries.active: per
mission denied"
panic: Unable to create mmap-ed active query log

goroutine 1 [running]:
github.com/prometheus/promql.NewActiveQueryTracker({0x1d79193, 0x1d}
, 0x14, {0x20745e0, 0xc00008d900})
    github.com/prometheus/prometheus/promql/query_logger.go:117 +0x3d7
main.main()
    github.com/prometheus/prometheus/cmd/prometheus/main.go:482 +0x6cbf
soriano2@server1-Server1:~$ prometheus --version
prometheus, version 2.31.2+ds1 (branch: debian/sid, revision: 2.31.2+ds1-1ubunt
u1)
    build user:      team+pkg-go@tracker.debian.org
    build date:      20220317-16:26:29
    go version:      go1.17.3
    platform:        linux/amd64
soriano2@server1-Server1:~$
```

- This shows the run of the prome.yml playbook file. The installation of prometheus on the ubuntu host/workstation is successful but the installation of prometheus for the CentOS is unsuccessful.



The image shows a terminal window titled 'Terminal' with the date 'Oct 27 09:42'. The user is 'soriano2' on a 'soriano2-localmachine'. The terminal displays the output of an Ansible playbook named 'prome.yml'. The output shows the installation of Prometheus for CentOS. The first task, 'Pre-req os installation for CentOS', is successful. The second task, 'Enabling snapd sockets for CentOS', is also successful. The third task, 'Installation of Prometheus for CentOS', is successful. The final output shows the status of the installation for two hosts: 192.168.56.105 and 192.168.56.118. The status for 192.168.56.105 is 'ok=4', 'changed=0', 'unreachable=0', and 'failed=0'. The status for 192.168.56.118 is 'ok=6', 'changed=2', 'unreachable=0', and 'failed=0'. The terminal also shows a 'PLAY RECAP' section.

```
*
skipping: [192.168.56.118]
ok: [192.168.56.105]

TASK [installprom : Pre-req os installation for CentOS] *****
*
skipping: [192.168.56.105]
ok: [192.168.56.118]

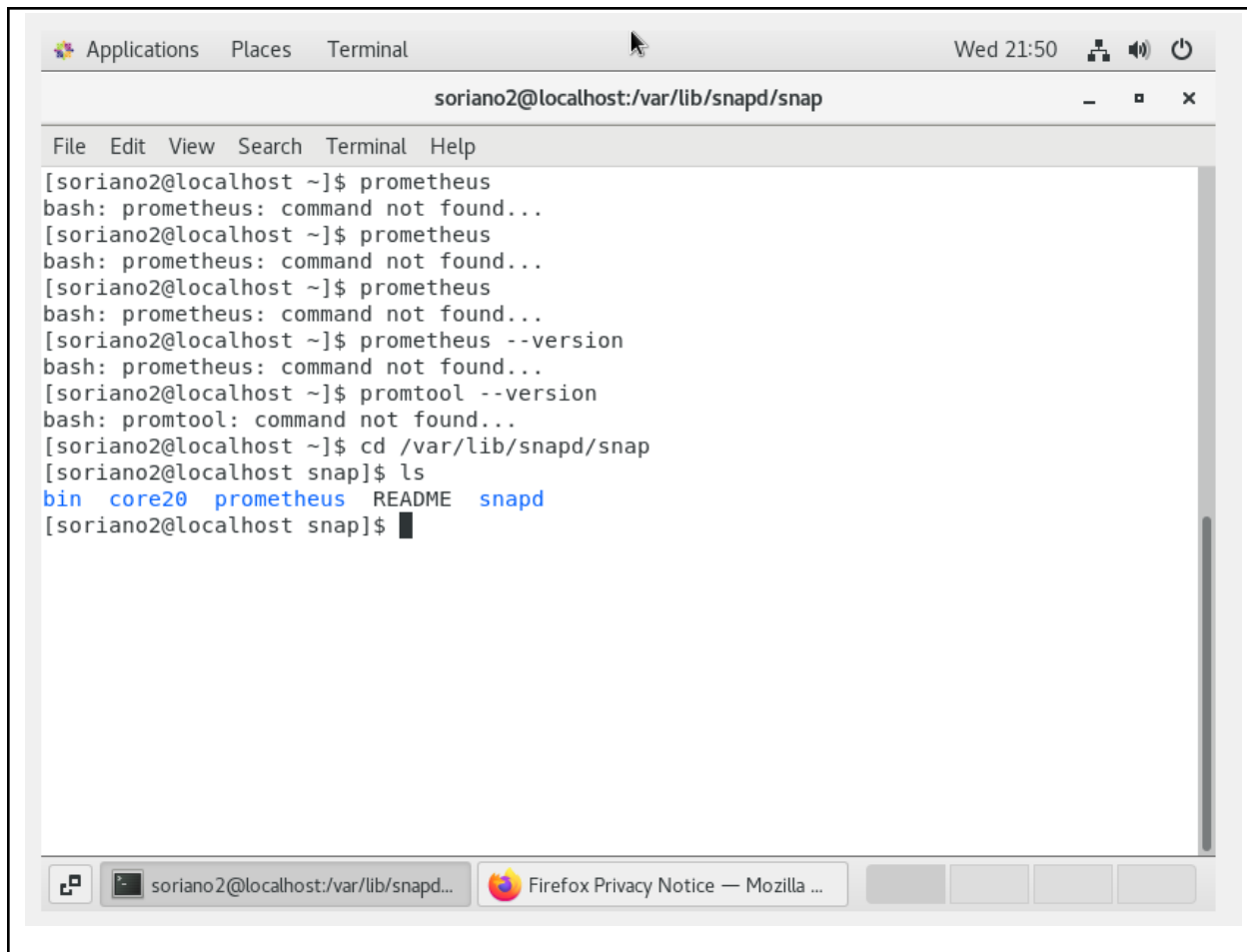
TASK [installprom : Enabling snapd sockets for CentOS] *****
*
skipping: [192.168.56.105]
changed: [192.168.56.118]

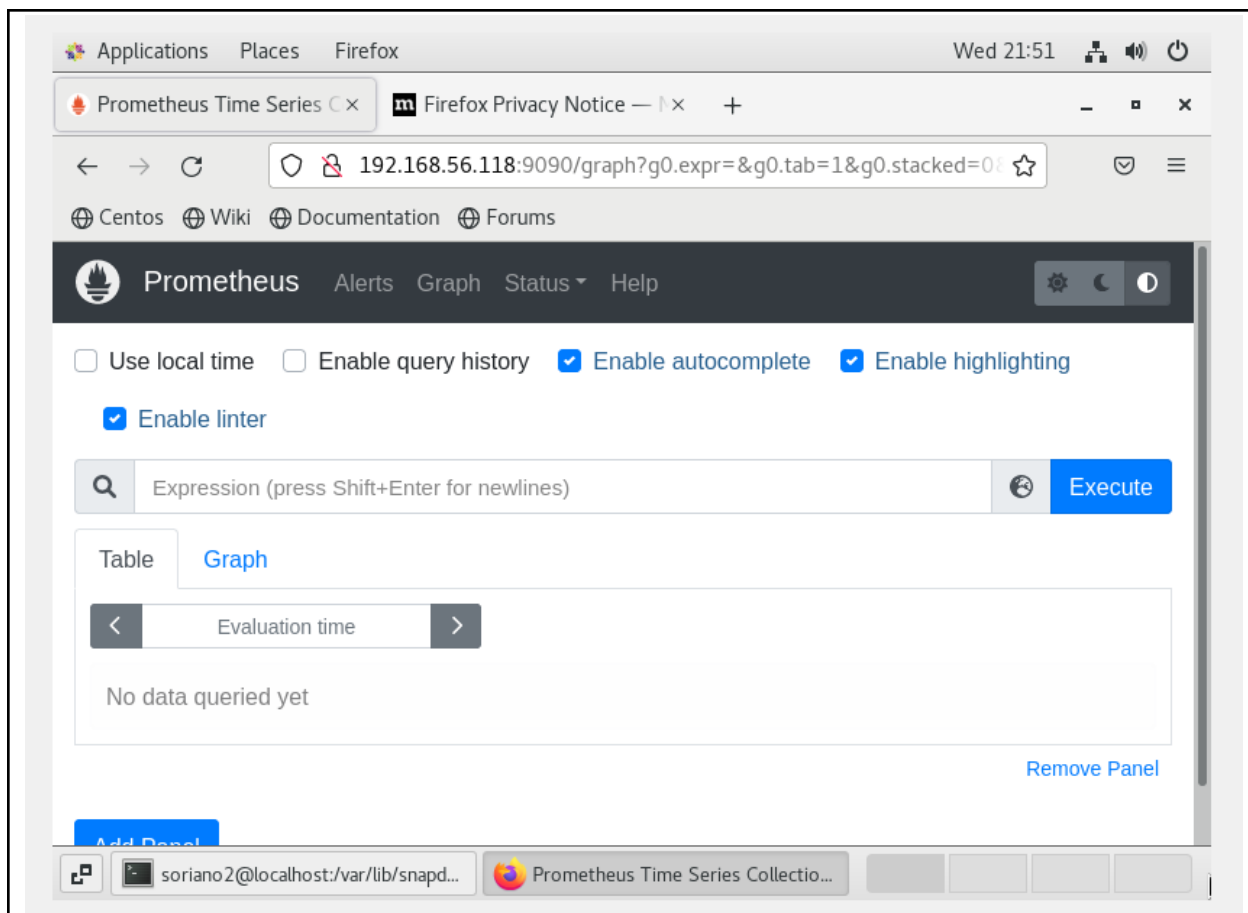
TASK [installprom : Installation of Prometheus for CentOS] *****
*
skipping: [192.168.56.105]
changed: [192.168.56.118]

PLAY RECAP *****
*
192.168.56.105      : ok=4    changed=0    unreachable=0    failed=0
skipped=4    rescued=0    ignored=0
192.168.56.118    : ok=6    changed=2    unreachable=0    failed=0
skipped=2    rescued=0    ignored=0

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

- This shows the re-run of the playbook file named prome.yml, this time the installation of prometheus to the CentOS workstation is now successful.





- This shows the prometheus proof of installation on the CentOS workstation.

Activities

Terminal

Oct 27 10:35



soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...



```
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$ ansible-playbook --ask-be  
come-pass infdb.yml  
BECOME password:
```



```
PLAY [all_servers] *****  
*
```



```
TASK [Gathering Facts] *****  
*  
ok: [192.168.56.118]  
ok: [192.168.56.105]
```



```
TASK [update repository index (CentOS)] *****  
*  
skipping: [192.168.56.105]  
ok: [192.168.56.118]
```



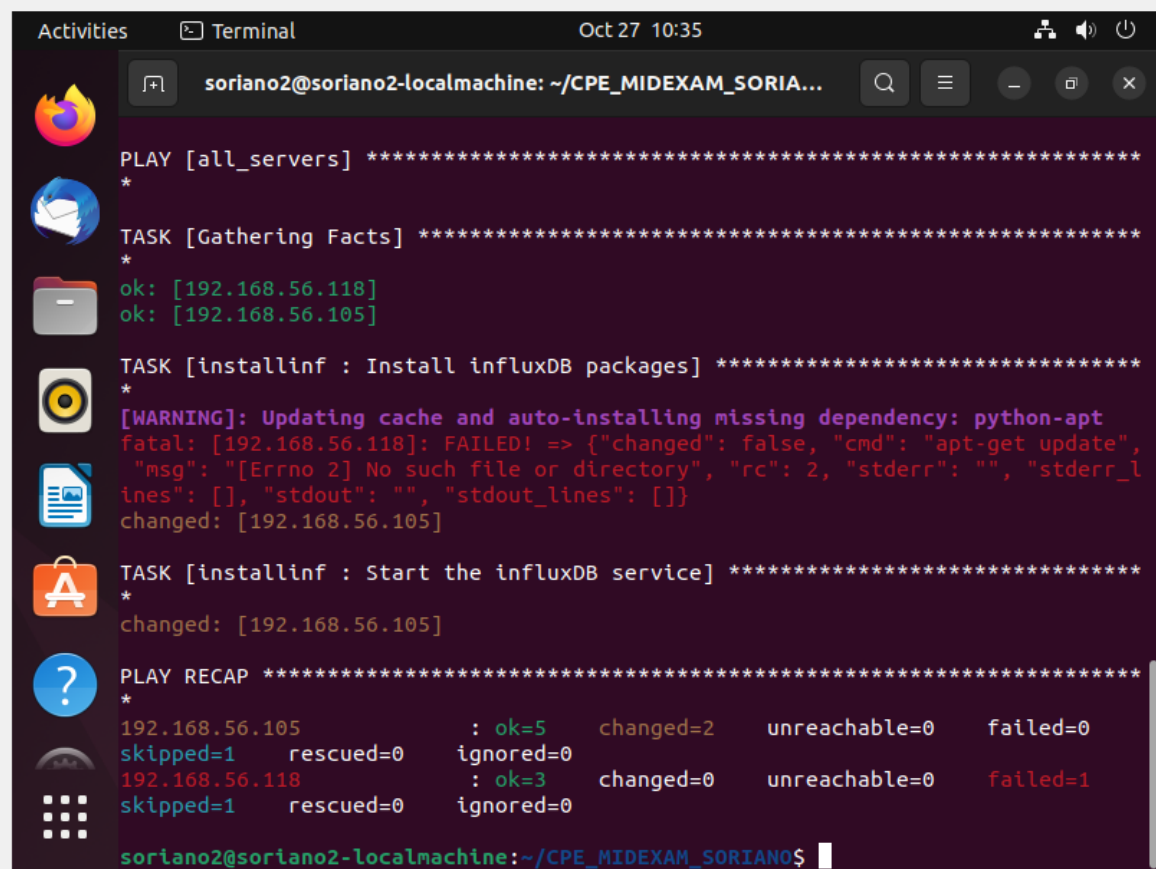
```
TASK [install updates (Ubuntu)] *****  
*  
skipping: [192.168.56.118]  
ok: [192.168.56.105]
```



```
PLAY [all_servers] *****  
*
```



```
TASK [Gathering Facts] *****  
*  
ok: [192.168.56.118]  
ok: [192.168.56.105]
```



The terminal window displays the execution of an Ansible playbook named 'infdb.yml'. The output is organized into sections for different tasks, with status icons on the left. The 'PLAY [all_servers]' section shows successful connections to both IP addresses. The 'TASK [Gathering Facts]' section shows successful fact gathering. The 'TASK [installinf : Install influxDB packages]' section shows a warning for the CentOS server (192.168.56.118) due to a missing dependency 'python-apt', which caused the installation to fail. The 'TASK [installinf : Start the influxDB service]' section shows successful service startup for both servers. The 'PLAY RECAP' section summarizes the results: 5 OK, 2 changed, 0 unreachable, 0 failed, 1 skipped, and 0 rescued for the Ubuntu server (192.168.56.105), and 3 OK, 0 changed, 0 unreachable, 1 failed, 1 skipped, and 0 rescued for the CentOS server (192.168.56.118).

```
Activities  Terminal  Oct 27 10:35
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...

PLAY [all_servers] *****
*

TASK [Gathering Facts] *****
*
ok: [192.168.56.118]
ok: [192.168.56.105]

TASK [installinf : Install influxDB packages] *****
*
[WARNING]: Updating cache and auto-installing missing dependency: python-apt
fatal: [192.168.56.118]: FAILED! => {"changed": false, "cmd": "apt-get update",
"msg": "[Errno 2] No such file or directory", "rc": 2, "stderr": "", "stderr_l
ines": [], "stdout": "", "stdout_lines": []}
changed: [192.168.56.105]

TASK [installinf : Start the influxDB service] *****
*
changed: [192.168.56.105]

PLAY RECAP *****
*
192.168.56.105      : ok=5    changed=2    unreachable=0    failed=0
skipped=1    rescued=0    ignored=0
192.168.56.118    : ok=3    changed=0    unreachable=0    failed=1
skipped=1    rescued=0    ignored=0

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

- This shows the run of the infdb.yml playbook file to install influxDB on both ubuntu and CentOS workstations. But there has been an error encountered on the installation process for the CentOS workstation.

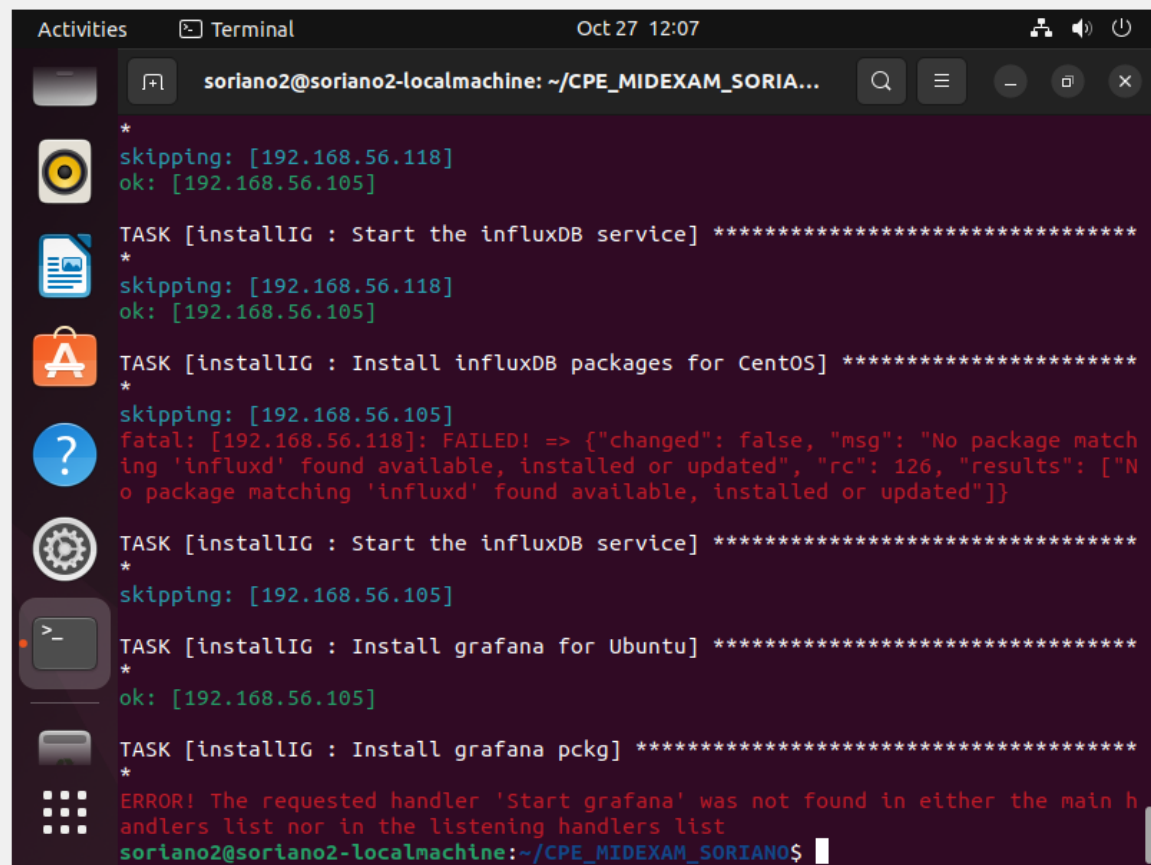
```

soriano2@server1-Server1: ~
soriano2@server1-Server1:~$ influxdb
Command 'influxdb' not found, did you mean:
  command 'influxd' from deb influxdb (1.6.7-rc0-1)
Try: sudo apt install <deb name>
soriano2@server1-Server1:~$ influxd

88888888      .d888 888      88888888b. 8888888b.
888      d88P" 888      888  "Y88b 888  "88b
888      888  888  888      888  888 888  .88P
888 88888b. 8888888 888 888 888 888 888 88888888K.
888 888 "88b 888 888 888 888 Y8bd8P" 888 888 888 "Y88b
888 888 888 888 888 888 888 X88K 888 888 888 888
888 888 888 888 888 Y88b 888 .d8""8b. 888 .d88P 888 d88P
88888888 888 888 888 888 "Y88888 888 888 88888888P" 88888888P"

2022-10-27T02:33:50.645040Z    info    InfluxDB starting    {"log_id": "0dm
pTDD0000", "version": "1.6.7-rc0", "branch": "unknown", "commit": "unknown"}
2022-10-27T02:33:50.645335Z    info    Go runtime    {"log_id": "0dmpTDD0000
", "version": "go1.15.4", "maxprocs": 1}
run: open server: listen: listen tcp 127.0.0.1:8088: bind: address already in u
se
soriano2@server1-Server1:~$
  
```

- **This is the proof that the influxDB packages have been successfully installed on the ubuntu workstation.**



The image shows a terminal window titled 'Terminal' with the date 'Oct 27 12:07'. The user is 'soriano2@soriano2-localmachine' and the current directory is '~/CPE_MIDEXAM_SORIA...'. The terminal output shows the following steps:

```
*
skipping: [192.168.56.118]
ok: [192.168.56.105]

TASK [installIG : Start the influxDB service] *****
*
skipping: [192.168.56.118]
ok: [192.168.56.105]

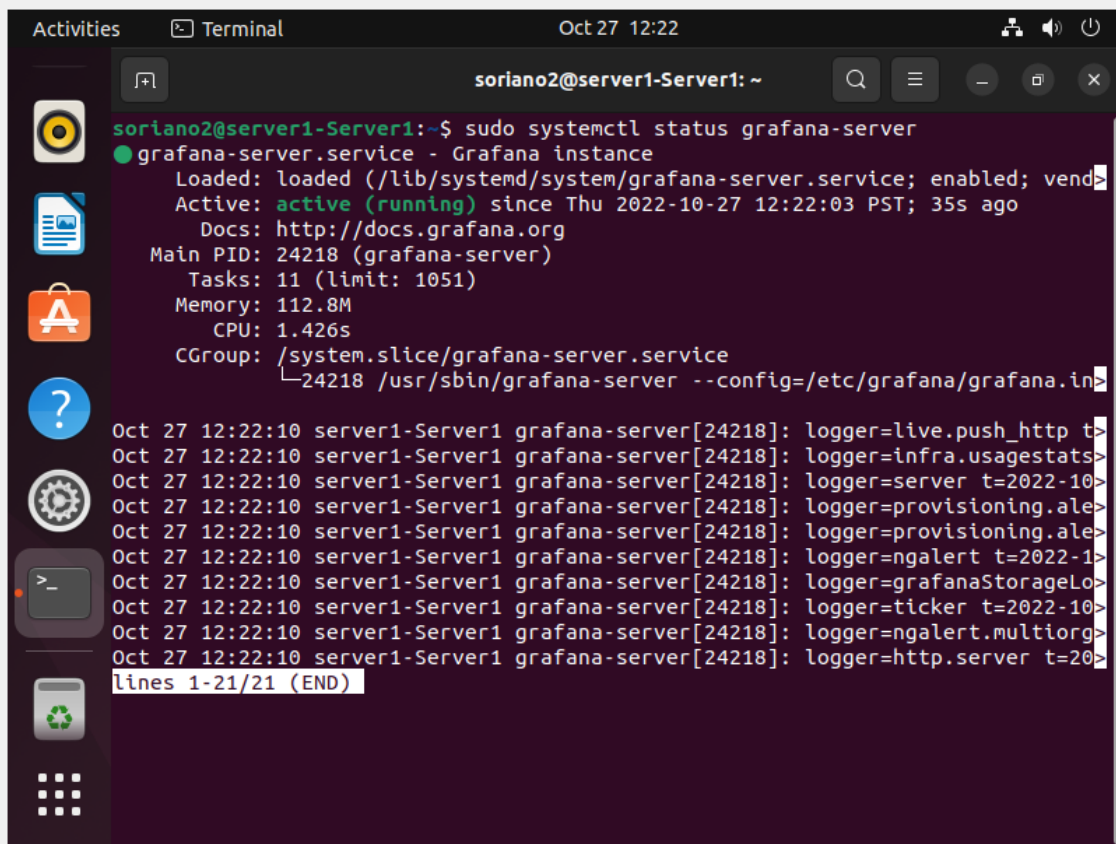
TASK [installIG : Install influxDB packages for CentOS] *****
*
skipping: [192.168.56.105]
fatal: [192.168.56.118]: FAILED! => {"changed": false, "msg": "No package match
ing 'influxd' found available, installed or updated", "rc": 126, "results": ["N
o package matching 'influxd' found available, installed or updated"]}

TASK [installIG : Start the influxDB service] *****
*
skipping: [192.168.56.105]

TASK [installIG : Install grafana for Ubuntu] *****
*
ok: [192.168.56.105]

TASK [installIG : Install grafana pkg] *****
*
ERROR! The requested handler 'Start grafana' was not found in either the main h
andlers list nor in the listening handlers list
soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

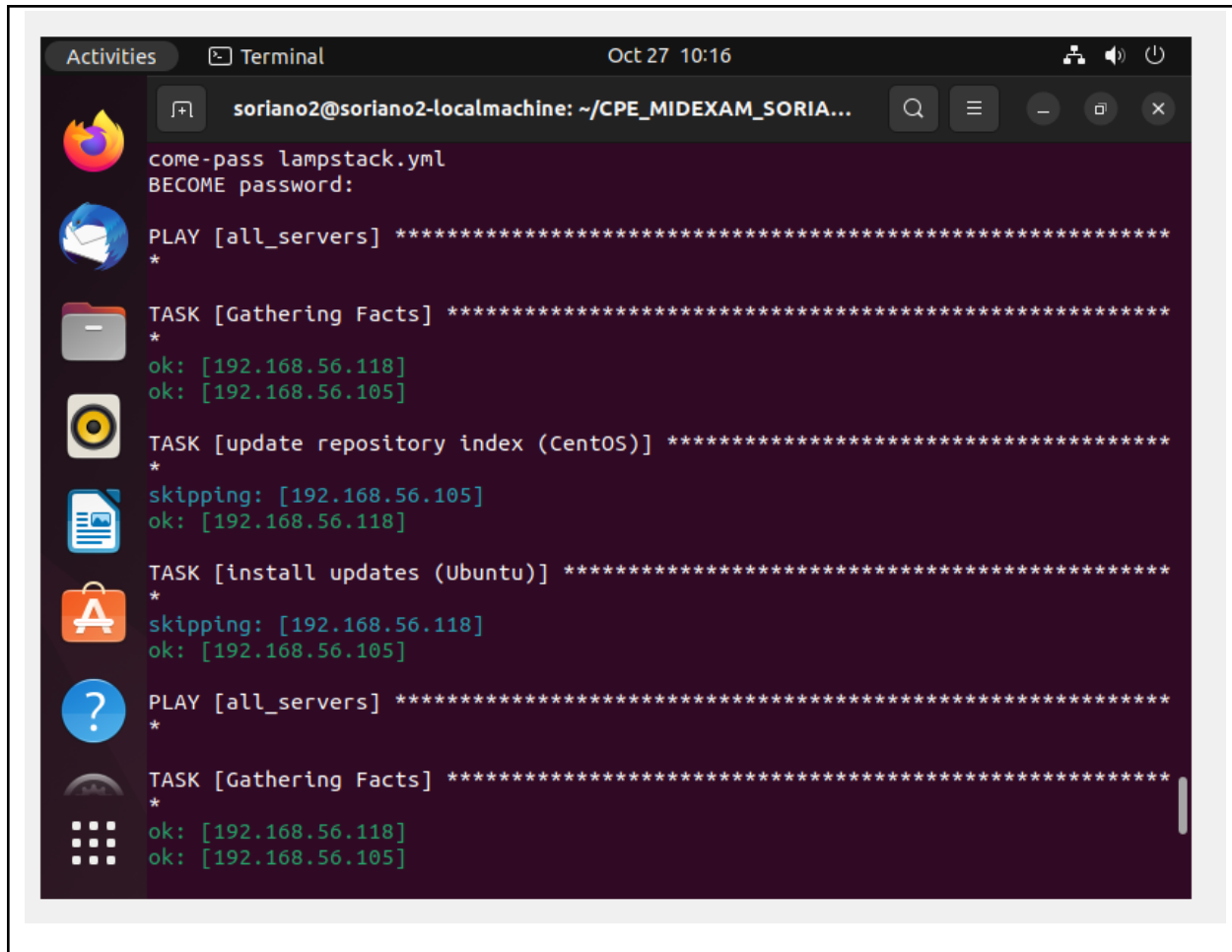
- This shows the successful installation of influxdb and grafana on the workstations.

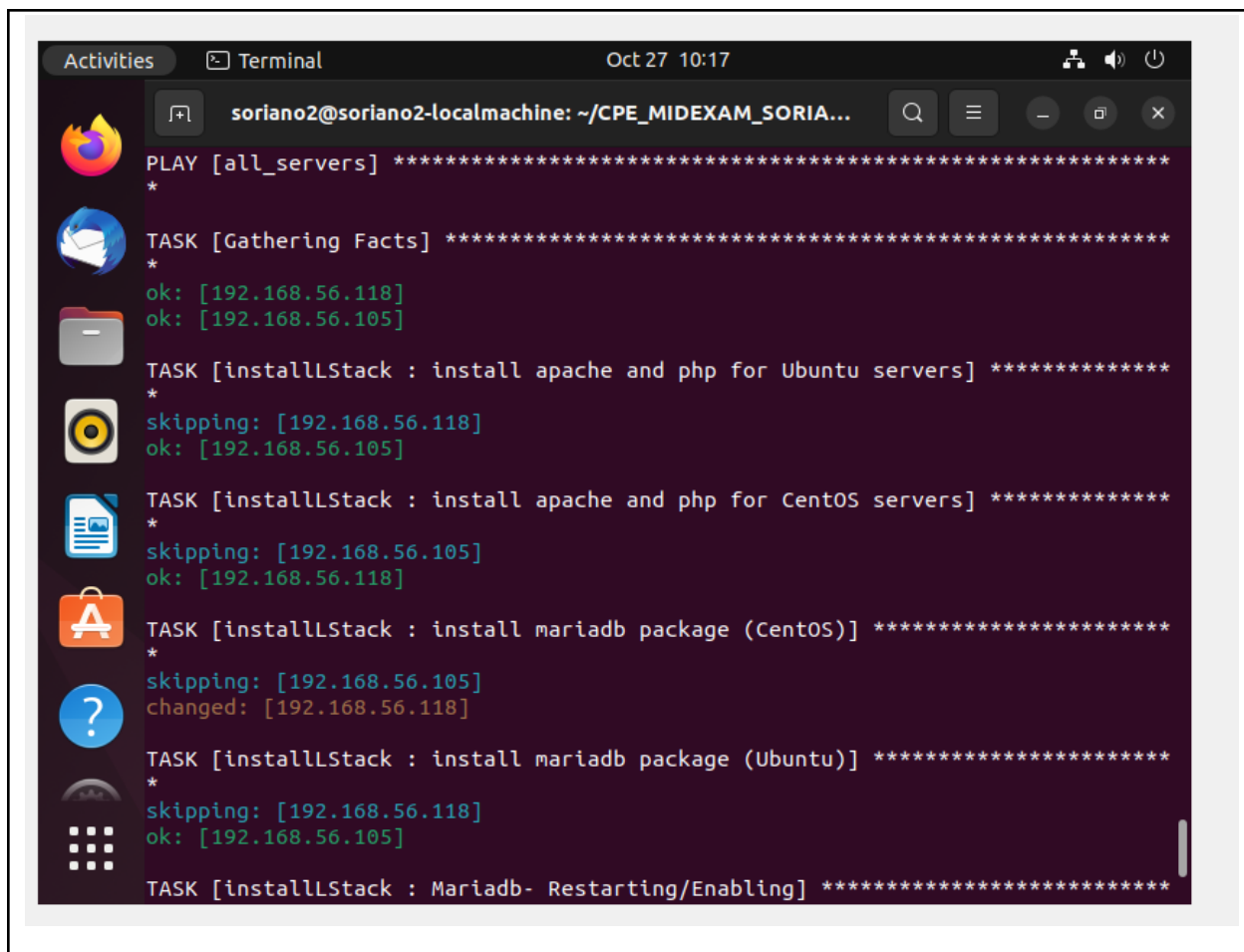
A terminal window titled 'Terminal' with a timestamp of 'Oct 27 12:22'. The user 'soriano2@server1-Server1: ~' has executed the command 'sudo systemctl status grafana-server'. The output shows the 'grafana-server.service' is 'active (running)' since 'Thu 2022-10-27 12:22:03 PST; 35s ago'. It lists various details like 'Main PID: 24218 (grafana-server)', 'Tasks: 11 (limit: 1051)', 'Memory: 112.8M', 'CPU: 1.426s', and 'CGroup: /system.slice/grafana-server.service'. Below this, a series of log messages from 'server1-Server1 grafana-server[24218]' are shown, including 'logger=live.push_http t>', 'logger=infra.usagestats>', 'logger=server t=2022-10>', 'logger=provisioning.ale>', 'logger=provisioning.ale>', 'logger=ngalert t=2022-1>', 'logger=grafanaStorageLo>', 'logger=ticker t=2022-10>', 'logger=ngalert.multiorg>', and 'logger=http.server t=20>'. The terminal also shows 'lines 1-21/21 (END)'.

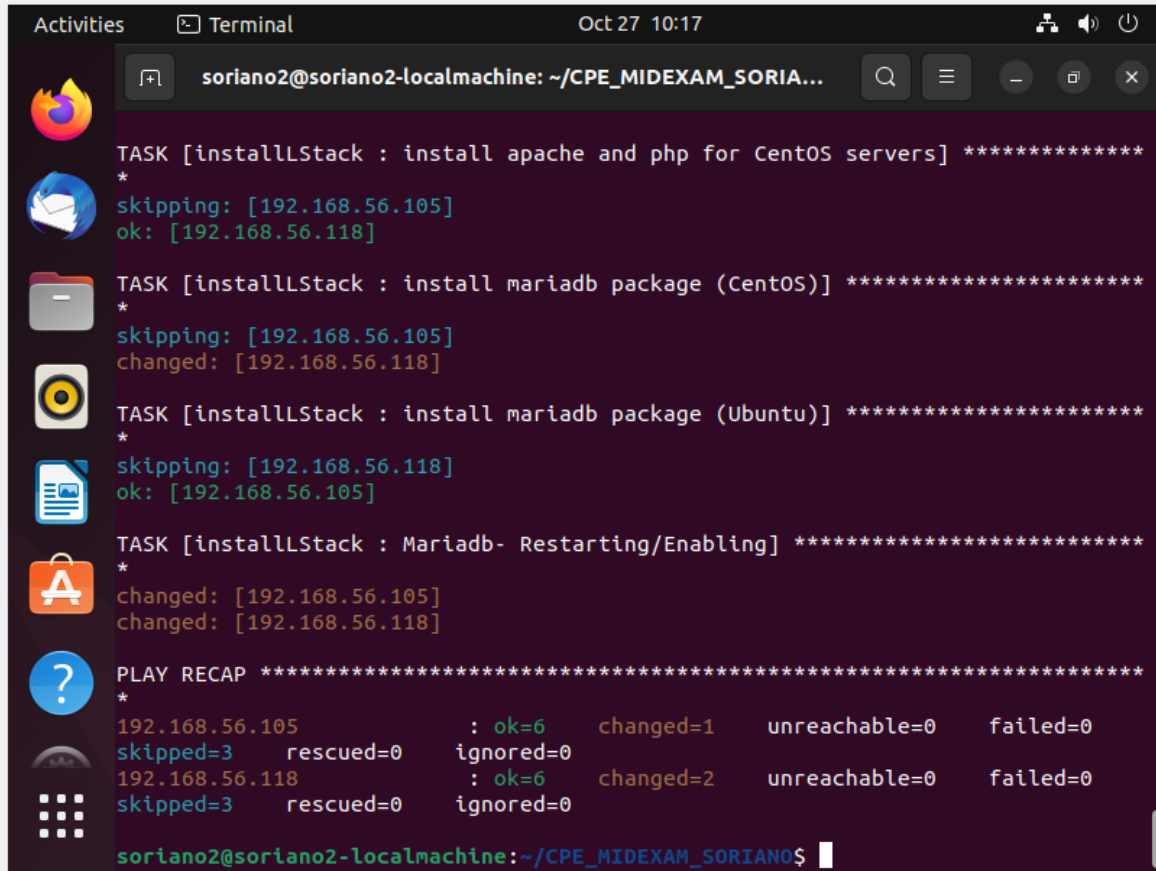
- This serves as the proof of installation of grafana on the workstations.

!2.4 SCREENSHOTS/PROOFS:

SCREENSHOTS:







```
Activities  Terminal  Oct 27 10:17
soriano2@soriano2-localmachine: ~/CPE_MIDEXAM_SORIA...

TASK [installLStack : install apache and php for CentOS servers] *****
*
skipping: [192.168.56.105]
ok: [192.168.56.118]

TASK [installLStack : install mariadb package (CentOS)] *****
*
skipping: [192.168.56.105]
changed: [192.168.56.118]

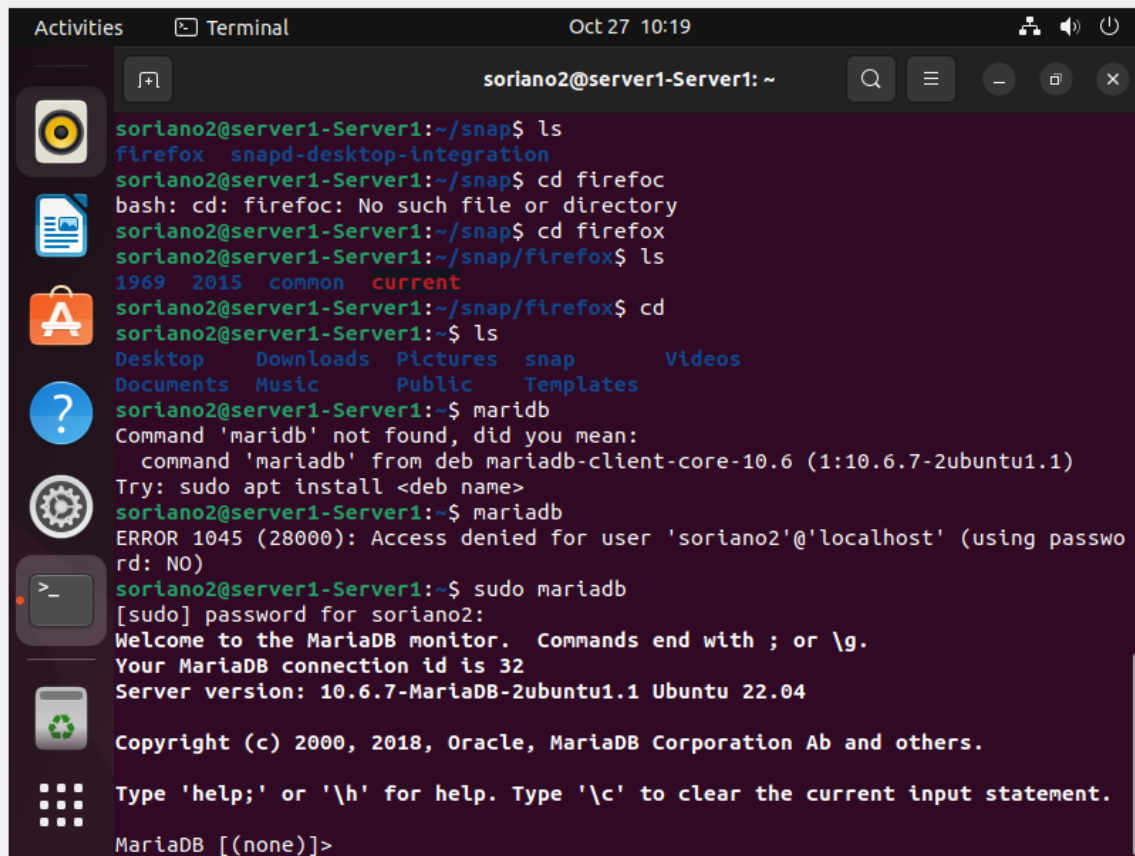
TASK [installLStack : install mariadb package (Ubuntu)] *****
*
skipping: [192.168.56.118]
ok: [192.168.56.105]

TASK [installLStack : Mariadb- Restarting/Enabling] *****
*
changed: [192.168.56.105]
changed: [192.168.56.118]

PLAY RECAP *****
*
192.168.56.105      : ok=6    changed=1    unreachable=0    failed=0
skipped=3    rescued=0    ignored=0
192.168.56.118    : ok=6    changed=2    unreachable=0    failed=0
skipped=3    rescued=0    ignored=0

soriano2@soriano2-localmachine:~/CPE_MIDEXAM_SORIANO$
```

- This shows the successful run of the lampstack.yml playbook file. As seen above, no errors occurred because the error has been resolved before running again the playbook.



The image shows a terminal window titled 'Terminal' with the date 'Oct 27 10:19'. The user 'soriano2' is logged in on 'server1-Server1'. The terminal output shows the following sequence of commands and responses:

```
soriano2@server1-Server1:~/snap$ ls
firefox  snapd-desktop-integration
soriano2@server1-Server1:~/snap$ cd firefoc
bash: cd: firefoc: No such file or directory
soriano2@server1-Server1:~/snap$ cd firefox
soriano2@server1-Server1:~/snap/firefox$ ls
1969  2015  common  current
soriano2@server1-Server1:~/snap/firefox$ cd
soriano2@server1-Server1:~$ ls
Desktop  Downloads  Pictures  snap      Videos
Documents  Music      Public   Templates
soriano2@server1-Server1:~$ maridb
Command 'maridb' not found, did you mean:
  command 'mariadb' from deb mariadb-client-core-10.6 (1:10.6.7-2ubuntu1.1)
Try: sudo apt install <deb name>
soriano2@server1-Server1:~$ mariadb
ERROR 1045 (28000): Access denied for user 'soriano2'@'localhost' (using password: NO)
soriano2@server1-Server1:~$ sudo mariadb
[sudo] password for soriano2:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

- This is the proof that the mariadb package installation on the ubuntu workstation is successful.

```
soriano2@localhost:~  
File Edit View Search Terminal Help  
[soriano2@localhost ~]$ mariadb --version  
bash: mariadb: command not found...  
[soriano2@localhost ~]$ mariadb-server  
bash: mariadb-server: command not found...  
[soriano2@localhost ~]$ ls  
Desktop Documents Downloads get-pip.py Music Pictures Public Templates Videos  
[soriano2@localhost ~]$ cd /var/lib/napd/snap  
[soriano2@localhost snap]$ ls  
bin core20 prometheus README snapd  
[soriano2@localhost snap]$ cd snapd  
[soriano2@localhost snapd]$ ls  
17336 current  
[soriano2@localhost snapd]$ cd  
[soriano2@localhost ~]$ cd  
[soriano2@localhost ~]$ cd  
[soriano2@localhost ~]$ sudo mysql  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 2  
Server version: 5.5.68-MariaDB MariaDB Server  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]>
```

- This is the proof that the mariadb package installation on the CentOS workstation is successful.

GitHub link:

[git@github.com:qgsoriano1/CPE_MIDEXAM_SORIANO.git](https://github.com/qgsoriano1/CPE_MIDEXAM_SORIANO.git)

Conclusions: (link your conclusion from the objective)

There are a lot of errors and unforeseen stuff while performing this examination. All of the requirements are quite achieved, which has proofs pasted above. The use of roles is also implemented in this exam and proofs can also be found above. The workflow is also achieved within this exam because the workflow of how the playbooks task's will be executed accordingly, because the playbook won't work fine if the workflow of the playbooks are not arranged properly. By this being said, I believe that all of the requirements here are achieved and performed.

Faculty Eval Part 2 SCREENSHOT:

