Name: Gabriel Soriano	**Date Performed: September 08, 202**
Course/Section: CPE232 - CPE31S23	**Date Submitted: September 14, 2022**
Instructor: Engr. Taylar	**Semester and SY: 1st sem/2022-2023**

Activity 4: Running Elevated Ad hoc Commands

1. Objectives:

1.1 Use commands that makes changes to remote machines

1.2 Use playbook in automating ansible commands

2. Discussion:

Provide screenshots for each task.

Elevated Ad hoc commands

So far, we have not performed ansible commands that makes changes to the remote servers. We manage to gather facts and connect to the remote machines, but we still did not make changes on those machines. In this activity, we will learn to use commands that would install, update, and upgrade packages in the remote machines. We will also create a playbook that will be used for automations.

Playbooks record and execute **Ansible**'s configuration, deployment, and orchestration functions. They can describe a policy you want your remote systems to enforce, or a set of steps in a general IT process. If Ansible modules are the tools in your workshop, playbooks are your instruction manuals, and your inventory of hosts are your raw material. At a basic level, playbooks can be used to manage configurations of and deployments to remote machines. At a more advanced level, they can sequence multi-tier rollouts involving rolling updates, and can delegate actions to other hosts, interacting with monitoring servers and load balancers along the way. You can check this documentation if you want to learn more about playbooks. Working with playbooks — Ansible Documentation

Task 1: Run elevated ad hoc commands

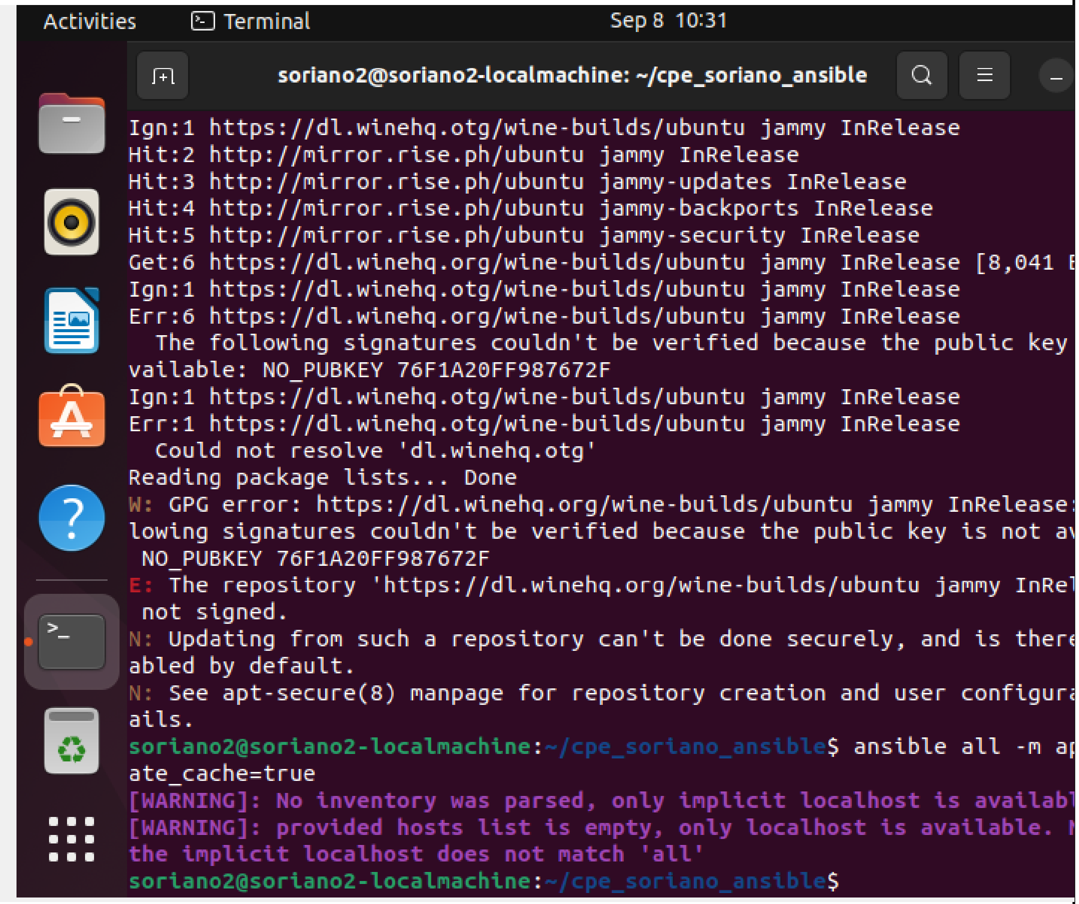
1. Locally, we use the command *sudo apt update* when we want to download package information from all configured resources. The sources often defined in /etc/apt/sources.list file and other files located in /etc/apt/sources.list.d/ directory. So, when you run update command, it downloads the package information from the Internet. It is useful to get info on an updated version of packages or their dependencies. We can only run

an apt update command in a remote machine. Issue the following command:

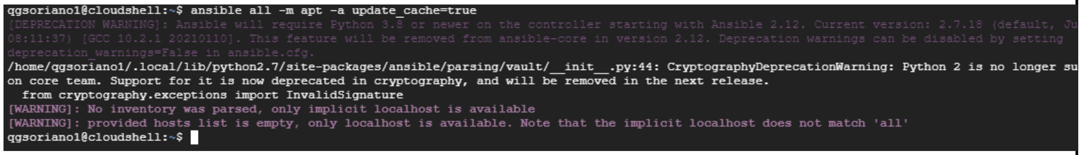
ansible all -m apt -a update_cache=true

What is the result of the command? Is it successful?

No, because something is still not one yet.

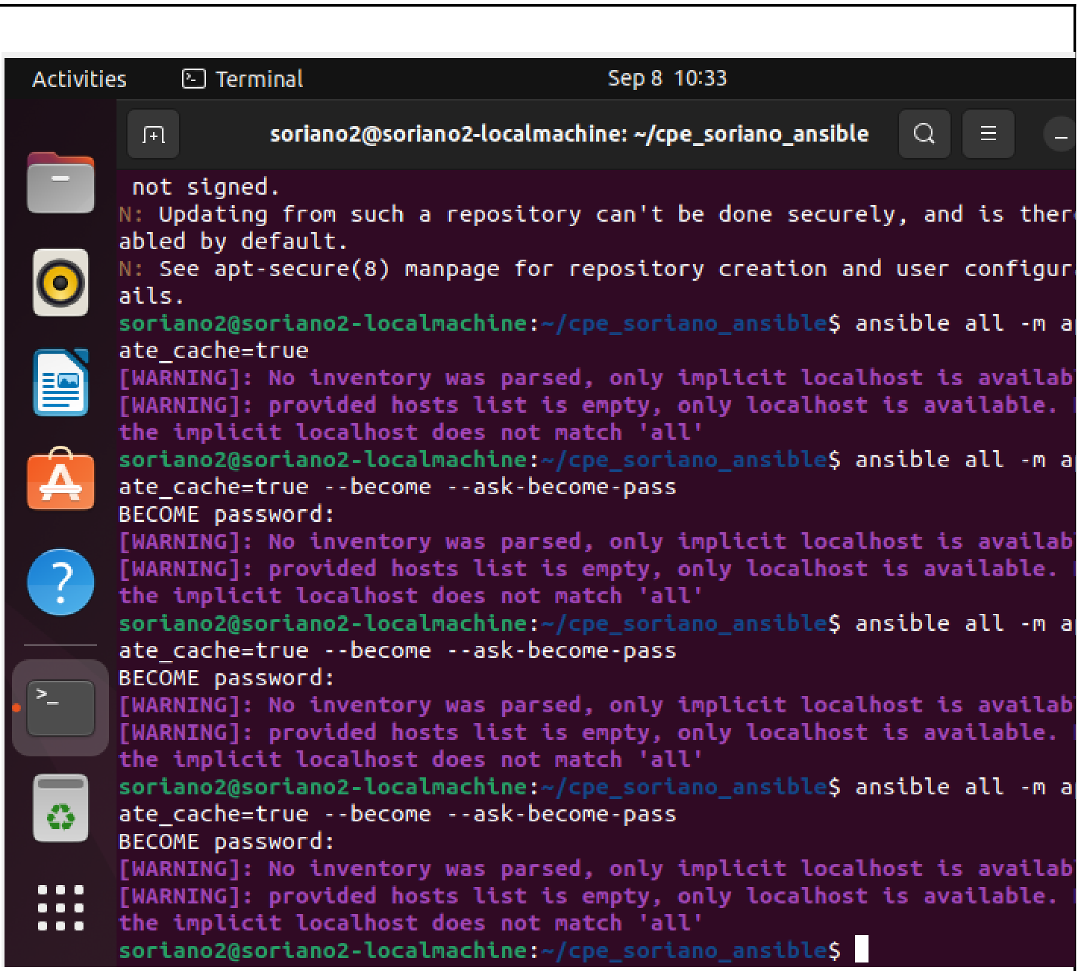


```
Ign:1 https://dl.winehq.otg/wine-builds/ubuntu jammy InRelease
Hit:2 http://mirror.rise.ph/ubuntu jammy InRelease
Hit:3 http://mirror.rise.ph/ubuntu jammy-updates InRelease
Hit:4 http://mirror.rise.ph/ubuntu jammy-backports InRelease
Hit:5 http://mirror.rise.ph/ubuntu jammy-security InRelease
Get:6 https://dl.winehq.org/wine-builds/ubuntu jammy InRelease [8,041 B
Ign:1 https://dl.winehq.otg/wine-builds/ubuntu jammy InRelease
Err:6 https://dl.winehq.org/wine-builds/ubuntu jammy InRelease
  The following signatures couldn't be verified because the public key
vailable: NO_PUBKEY 76F1A20FF987672F
Ign:1 https://dl.winehq.otg/wine-builds/ubuntu jammy InRelease
Err:1 https://dl.winehq.otg/wine-builds/ubuntu jammy InRelease
  Could not resolve 'dl.winehq.otg'
Reading package lists... Done
W: GPG error: https://dl.winehq.org/wine-builds/ubuntu jammy InRelease:
lowing signatures couldn't be verified because the public key is not av
 NO_PUBKEY 76F1A20FF987672F
E: The repository 'https://dl.winehq.org/wine-builds/ubuntu jammy InRel
 not signed.
N: Updating from such a repository can't be done securely, and is there
abled by default.
N: See apt-secure(8) manpage for repository creation and user configura
ails.
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -m ap
ate_cache=true
[WARNING]: No inventory was parsed, only implicit localhost is availabl
[WARNING]: provided hosts list is empty, only localhost is available. N
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$
```

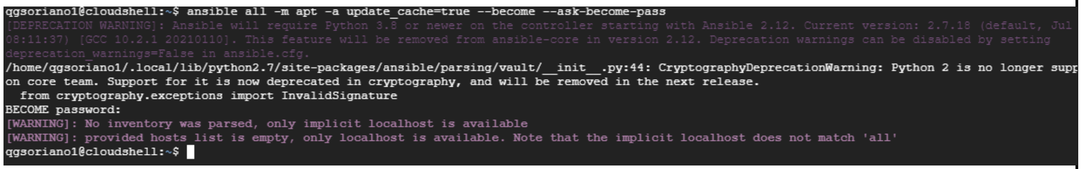


```
qgsoriano1@cloudshell:~$ ansible all -m apt -a update_cache=true
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (default, Ju
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no longer su
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
qgsoriano1@cloudshell:~$
```

Try editing the command and add something that would elevate the privilege. Issue the command *ansible all -m apt -a update_cache=true --become --ask-become-pass.* Enter the sudo password when prompted. You will notice now that the output of this command is a success. The *update_cache=true* is the same thing as running *sudo apt update*. The --become command elevate the privileges and the *--ask-become-pass* asks for the password. For now, even if we only have changed the packaged index, we were able to change something on the remote server.



```
 not signed.
N: Updating from such a repository can't be done securely, and is there
abled by default.
N: See apt-secure(8) manpage for repository creation and user configur
ails.
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -m a
ate_cache=true
[WARNING]: No inventory was parsed, only implicit localhost is availab
[WARNING]: provided hosts list is empty, only localhost is available.
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -m a
ate_cache=true --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availab
[WARNING]: provided hosts list is empty, only localhost is available.
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -m a
ate_cache=true --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availab
[WARNING]: provided hosts list is empty, only localhost is available.
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -m a
ate_cache=true --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availab
[WARNING]: provided hosts list is empty, only localhost is available.
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ 
```

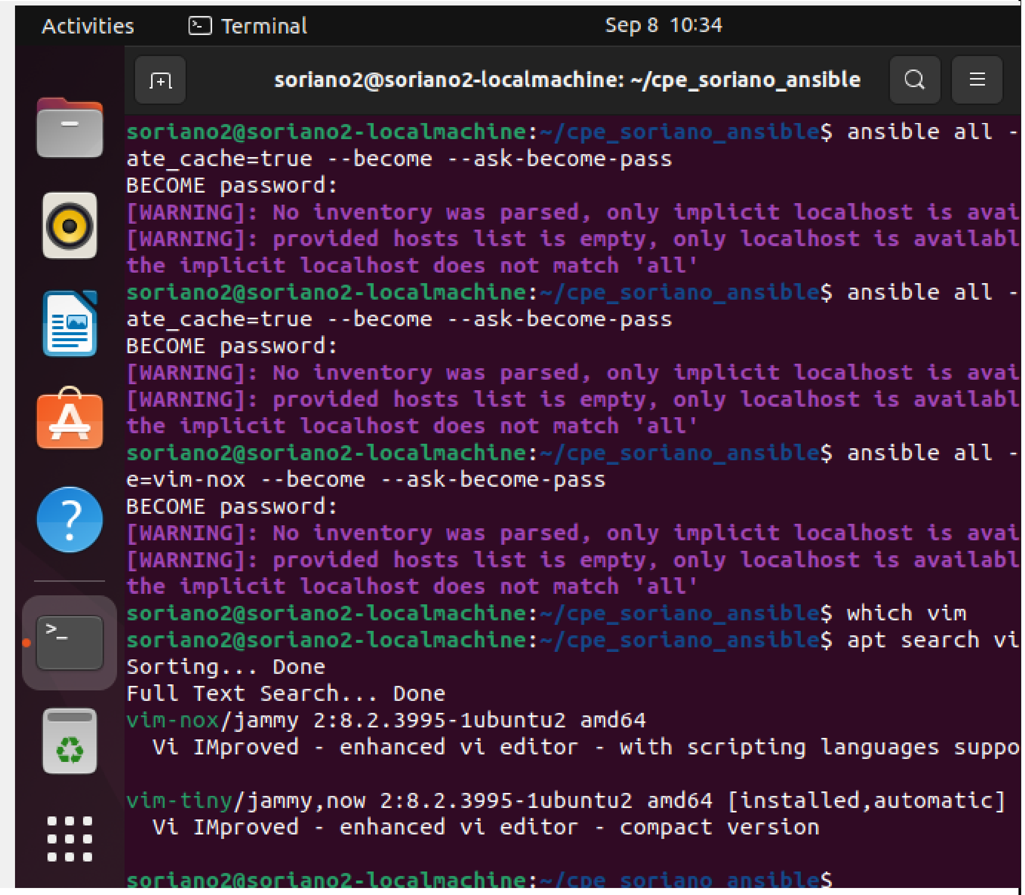


```
qgsoriano1@cloudshell:~$ ansible all -m apt -a update_cache=true --become --ask-become-pass
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (default, Jul
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no longer supp
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
qgsoriano1@cloudshell:~$ 
```

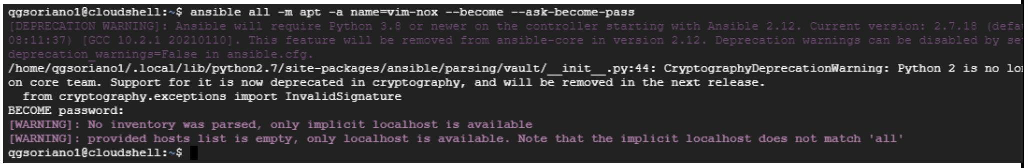
You may notice after the second command was executed, the status is CHANGED compared to the first command, which is FAILED.

2. Let's try to install VIM, which is an almost compatible version of the UNIX editor Vi. To do this, we will just changed the module part in 1.1 instruction. Here is the command: *ansible all -m apt -a name=vim-nox --become --ask-become-pass.* The command would take some time after typing the password because the local machine instructed the remote servers to actually install the package.
   2.1 Verify that you have installed the package in the remote servers. Issue the command *which vim* and the command *apt search vim-nox* respectively. Was the command successful?

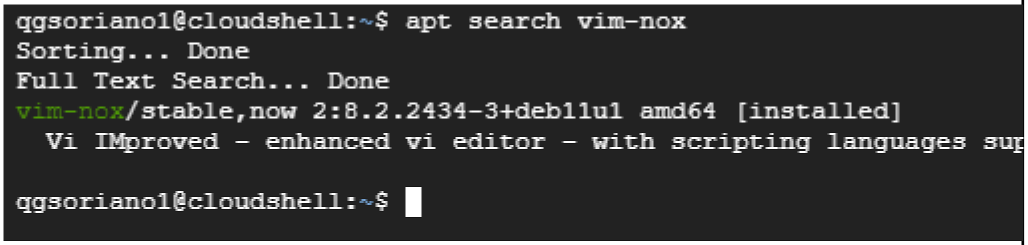
I think yes, because the results shows that the sorting and searching is done.



```
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -
ate_cache=true --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is avai
[WARNING]: provided hosts list is empty, only localhost is availabl
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -
ate_cache=true --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is avai
[WARNING]: provided hosts list is empty, only localhost is availabl
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ ansible all -
e=vim-nox --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is avai
[WARNING]: provided hosts list is empty, only localhost is availabl
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ which vim
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ apt search vi
Sorting... Done
Full Text Search... Done
vim-nox/jammy 2:8.2.3995-1ubuntu2 amd64
  Vi IMproved - enhanced vi editor - with scripting languages suppo

vim-tiny/jammy,now 2:8.2.3995-1ubuntu2 amd64 [installed,automatic]
  Vi IMproved - enhanced vi editor - compact version

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



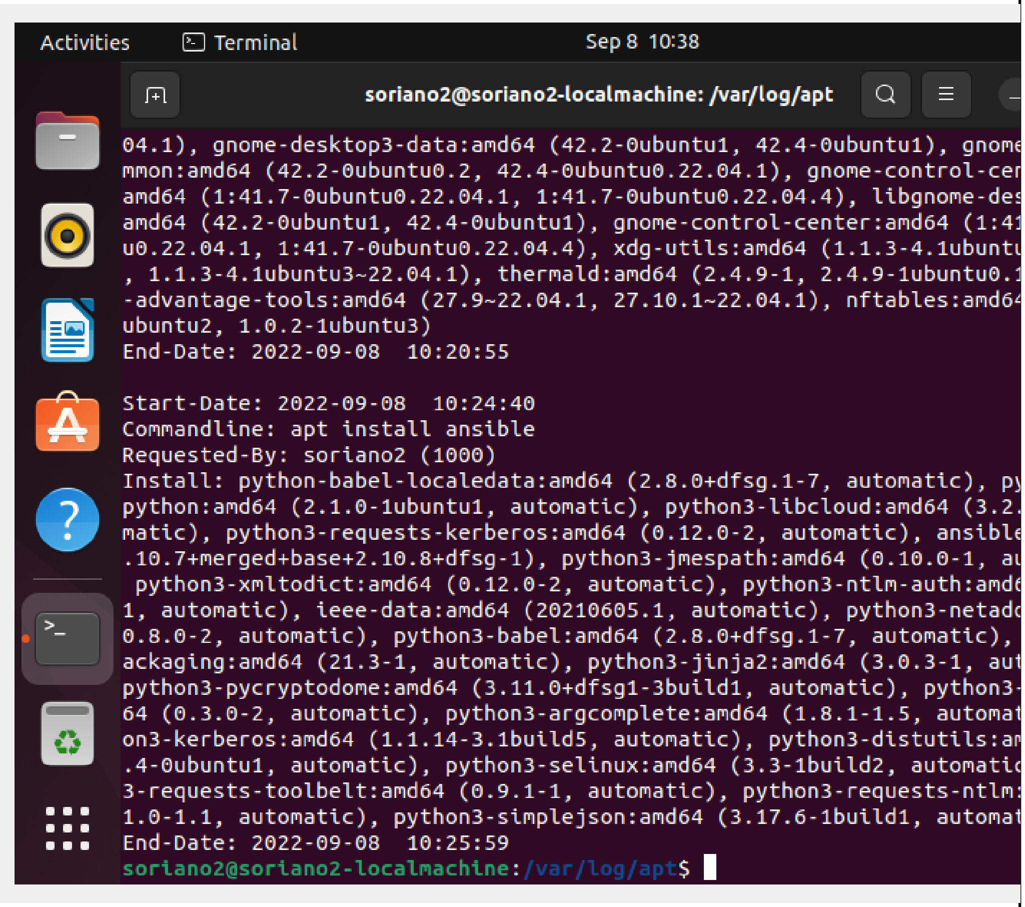
```
qgsoriano1@cloudshell:~$ ansible all -m apt -a name=vim-nox --become --ask-become-pass
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (defa
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by se
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no lo
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
qgsoriano1@cloudshell:~$
```



```
qgsoriano1@cloudshell:~$ apt search vim-nox
Sorting... Done
Full Text Search... Done
vim-nox/stable,now 2:8.2.2434-3+deb11u1 amd64 [installed]
  Vi IMproved - enhanced vi editor - with scripting languages sup

qgsoriano1@cloudshell:~$
```

2.2 Check the logs in the servers using the following commands: *cd /var/log*. After this, issue the command *ls,* go to the folder *apt* and open history.log. Describe what you see in the history.log.

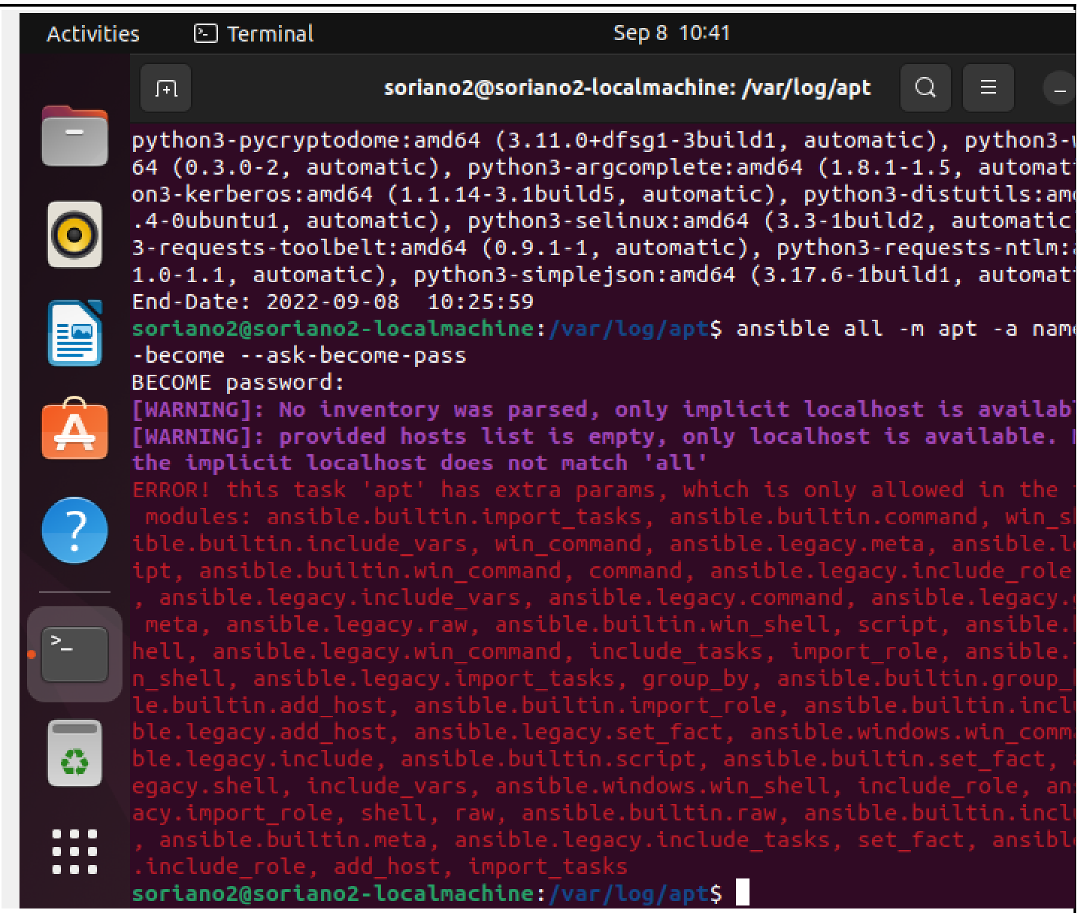
After executing the command open the history.log, there are logs that are recorded like the command line, the start date, and also the user that is requesting that specific command.



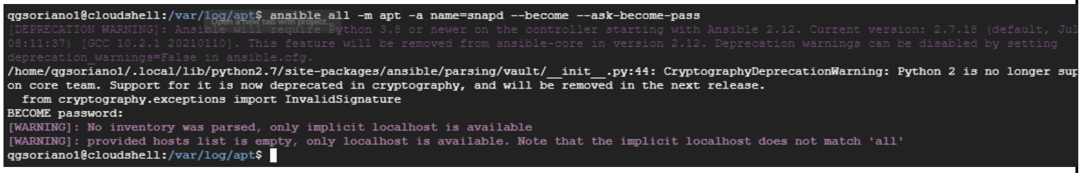
```
04.1), gnome-desktop3-data:amd64 (42.2-0ubuntu1, 42.4-0ubuntu1), gnome
mmon:amd64 (42.2-0ubuntu0.2, 42.4-0ubuntu0.22.04.1), gnome-control-cer
amd64 (1:41.7-0ubuntu0.22.04.1, 1:41.7-0ubuntu0.22.04.4), libgnome-des
amd64 (42.2-0ubuntu1, 42.4-0ubuntu1), gnome-control-center:amd64 (1:41
u0.22.04.1, 1:41.7-0ubuntu0.22.04.4), xdg-utils:amd64 (1.1.3-4.1ubuntu
, 1.1.3-4.1ubuntu3~22.04.1), thermald:amd64 (2.4.9-1, 2.4.9-1ubuntu0.1
-advantage-tools:amd64 (27.9~22.04.1, 27.10.1~22.04.1), nftables:amd64
ubuntu2, 1.0.2-1ubuntu3)
End-Date: 2022-09-08  10:20:55

Start-Date: 2022-09-08  10:24:40
Commandline: apt install ansible
Requested-By: soriano2 (1000)
Install: python-babel-localedata:amd64 (2.8.0+dfsg.1-7, automatic), py
python:amd64 (2.1.0-1ubuntu1, automatic), python3-libcloud:amd64 (3.2.
matic), python3-requests-kerberos:amd64 (0.12.0-2, automatic), ansible
.10.7+merged+base+2.10.8+dfsg-1), python3-jmespath:amd64 (0.10.0-1, au
 python3-xmltodict:amd64 (0.12.0-2, automatic), python3-ntlm-auth:amd6
1, automatic), ieee-data:amd64 (20210605.1, automatic), python3-netadd
0.8.0-2, automatic), python3-babel:amd64 (2.8.0+dfsg.1-7, automatic),
ackaging:amd64 (21.3-1, automatic), python3-jinja2:amd64 (3.0.3-1, aut
python3-pycryptodome:amd64 (3.11.0+dfsg1-3build1, automatic), python3-
64 (0.3.0-2, automatic), python3-argcomplete:amd64 (1.8.1-1.5, automat
on3-kerberos:amd64 (1.1.14-3.1build5, automatic), python3-distutils:am
.4-0ubuntu1, automatic), python3-selinux:amd64 (3.3-1build2, automatic
3-requests-toolbelt:amd64 (0.9.1-1, automatic), python3-requests-ntlm:
1.0-1.1, automatic), python3-simplejson:amd64 (3.17.6-1build1, automat
End-Date: 2022-09-08  10:25:59
soriano2@soriano2-localmachine:/var/log/apt$ 
```

3. This time, we will install a package called snapd. Snap is pre-installed in Ubuntu system. However, our goal is to create a command that checks for the latest installation package.

   3.1 Issue the command: *ansible all -m apt -a name=snapd --become --ask-become-pass*
   Can you describe the result of this command? Is it a success? Did it change anything in the remote servers?
   It is not successful. This error shows that the task 'apt' has extra parameters, which is only allowed in the specific list of modules.



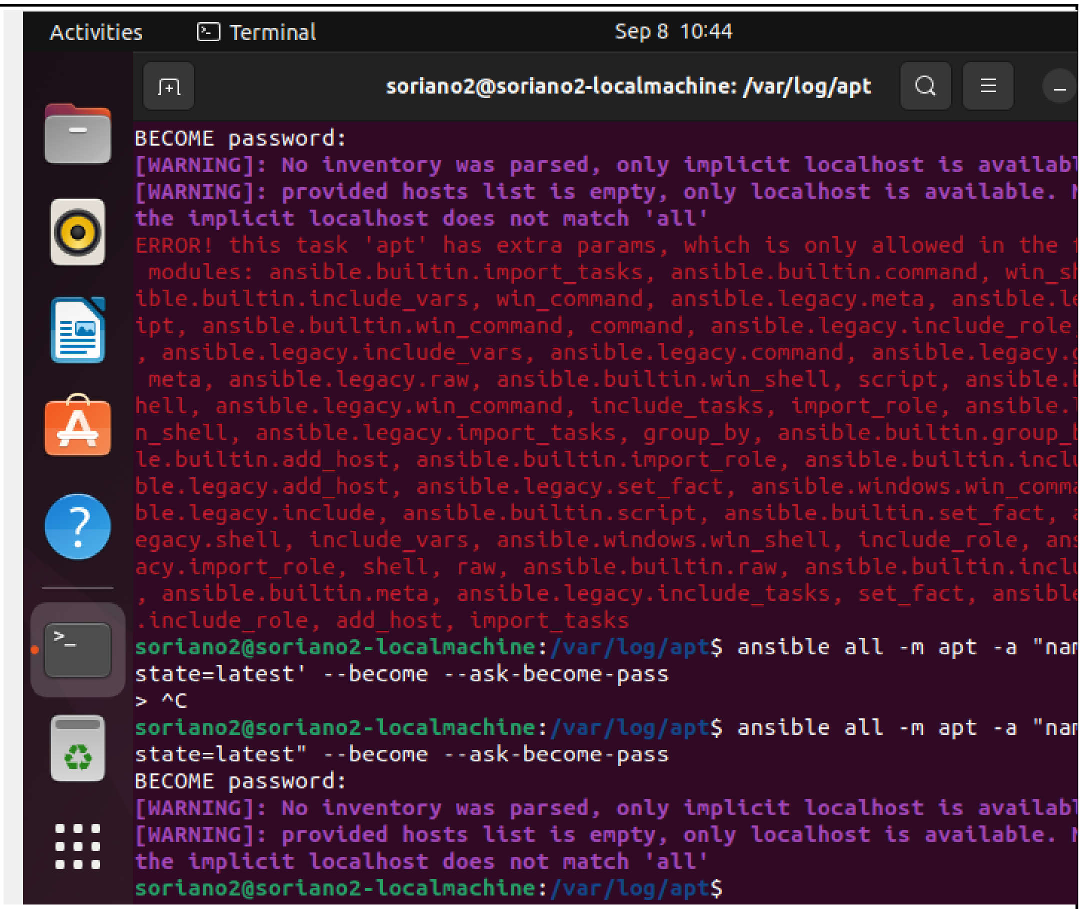
```
python3-pycryptodome:amd64 (3.11.0+dfsg1-3build1, automatic), python3-
64 (0.3.0-2, automatic), python3-argcomplete:amd64 (1.8.1-1.5, automat
on3-kerberos:amd64 (1.1.14-3.1build5, automatic), python3-distutils:am
.4-0ubuntu1, automatic), python3-selinux:amd64 (3.3-1build2, automatic
3-requests-toolbelt:amd64 (0.9.1-1, automatic), python3-requests-ntlm:
1.0-1.1, automatic), python3-simplejson:amd64 (3.17.6-1build1, automat
End-Date: 2022-09-08  10:25:59
soriano2@soriano2-localmachine:/var/log/apt$ ansible all -m apt -a nam
-become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availab
[WARNING]: provided hosts list is empty, only localhost is available. 
the implicit localhost does not match 'all'
ERROR! this task 'apt' has extra params, which is only allowed in the 
 modules: ansible.builtin.import_tasks, ansible.builtin.command, win_s
ible.builtin.include_vars, win_command, ansible.legacy.meta, ansible.l
ipt, ansible.builtin.win_command, command, ansible.legacy.include_role
, ansible.legacy.include_vars, ansible.legacy.command, ansible.legacy.
 meta, ansible.legacy.raw, ansible.builtin.win_shell, script, ansible.
hell, ansible.legacy.win_command, include_tasks, import_role, ansible.
n_shell, ansible.legacy.import_tasks, group_by, ansible.builtin.group_
le.builtin.add_host, ansible.builtin.import_role, ansible.builtin.incl
ble.legacy.add_host, ansible.legacy.set_fact, ansible.windows.win_comm
ble.legacy.include, ansible.builtin.script, ansible.builtin.set_fact, 
egacy.shell, include_vars, ansible.windows.win_shell, include_role, an
acy.import_role, shell, raw, ansible.builtin.raw, ansible.builtin.incl
, ansible.builtin.meta, ansible.legacy.include_tasks, set_fact, ansibl
.include_role, add_host, import_tasks
soriano2@soriano2-localmachine:/var/log/apt$ 
```



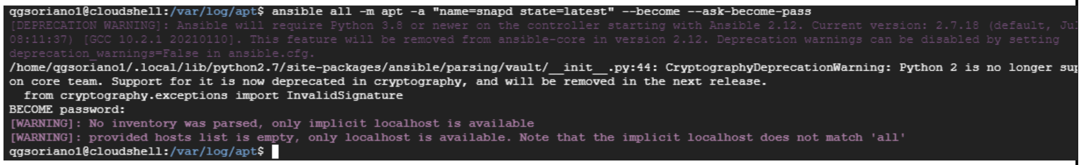
```
qgsoriano1@cloudshell:/var/log/apt$ ansible all -m apt -a name=snapd --become --ask-become-pass
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (default, Ju
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no longer sup
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
qgsoriano1@cloudshell:/var/log/apt$ 
```

3.2 Now, try to issue this command: *ansible all -m apt -a "name=snapd state=latest" --become --ask-become-pass*

Describe the output of this command. Notice how we added the command *state=latest* and placed them in double quotations.

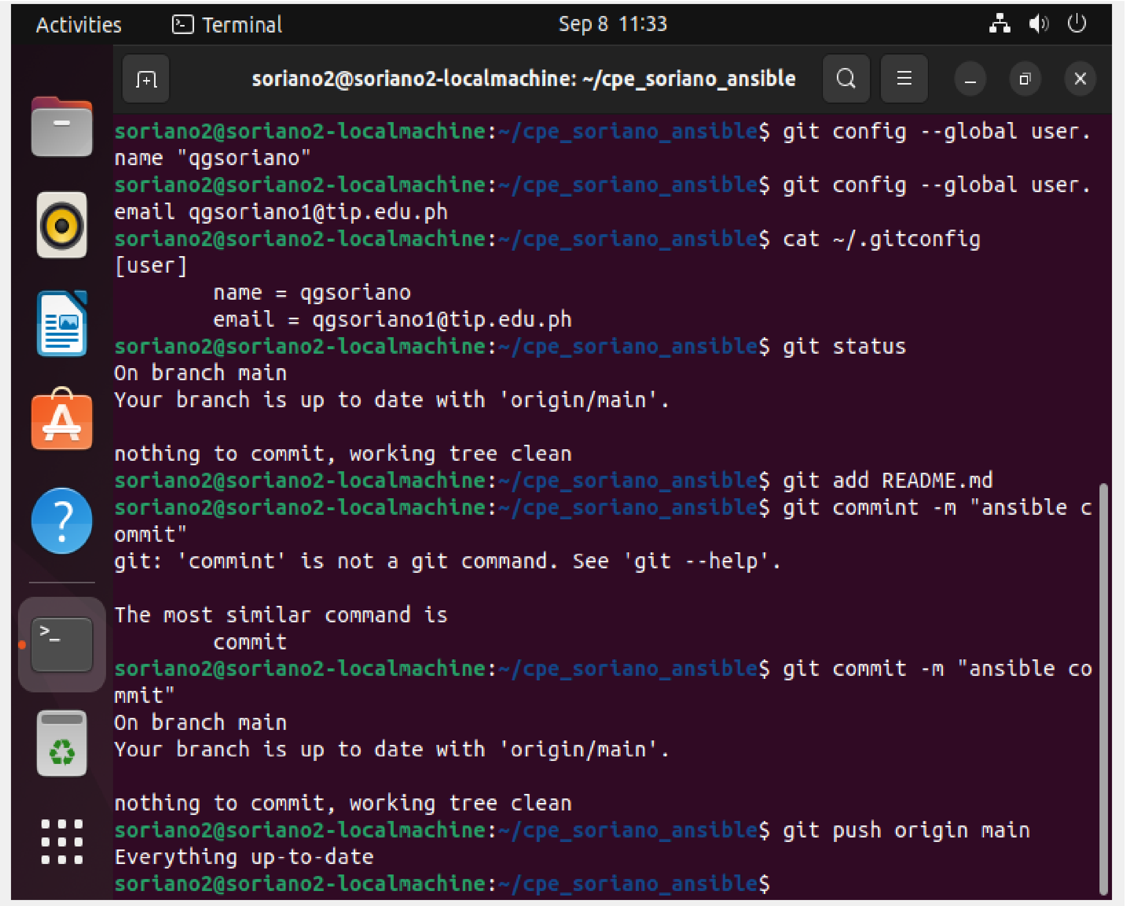


```
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availabl
[WARNING]: provided hosts list is empty, only localhost is available. N
the implicit localhost does not match 'all'
ERROR! this task 'apt' has extra params, which is only allowed in the f
 modules: ansible.builtin.import_tasks, ansible.builtin.command, win_sh
ible.builtin.include_vars, win_command, ansible.legacy.meta, ansible.le
ipt, ansible.builtin.win_command, command, ansible.legacy.include_role
, ansible.legacy.include_vars, ansible.legacy.command, ansible.legacy.g
 meta, ansible.legacy.raw, ansible.builtin.win_shell, script, ansible.b
hell, ansible.legacy.win_command, include_tasks, import_role, ansible.l
n_shell, ansible.legacy.import_tasks, group_by, ansible.builtin.group_b
le.builtin.add_host, ansible.builtin.import_role, ansible.builtin.inclu
ble.legacy.add_host, ansible.legacy.set_fact, ansible.windows.win_comma
ble.legacy.include, ansible.builtin.script, ansible.builtin.set_fact, a
egacy.shell, include_vars, ansible.windows.win_shell, include_role, ans
acy.import_role, shell, raw, ansible.builtin.raw, ansible.builtin.inclu
, ansible.builtin.meta, ansible.legacy.include_tasks, set_fact, ansible
.include_role, add_host, import_tasks
soriano2@soriano2-localmachine:/var/log/apt$ ansible all -m apt -a "nam
state=latest' --become --ask-become-pass
> ^C
soriano2@soriano2-localmachine:/var/log/apt$ ansible all -m apt -a "nam
state=latest" --become --ask-become-pass
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is availabl
[WARNING]: provided hosts list is empty, only localhost is available. N
the implicit localhost does not match 'all'
soriano2@soriano2-localmachine:/var/log/apt$
```

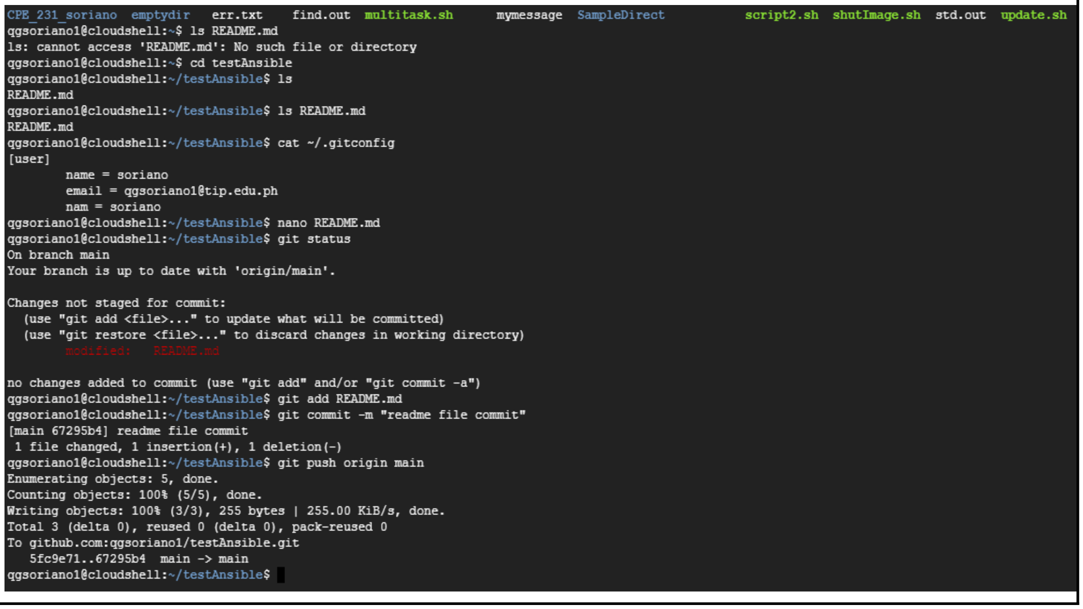


```
qgsoriano1@cloudshell:/var/log/apt$ ansible all -m apt -a "name=snapd state=latest" --become --ask-become-pass
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (default, Ju
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no longer sup
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
qgsoriano1@cloudshell:/var/log/apt$
```

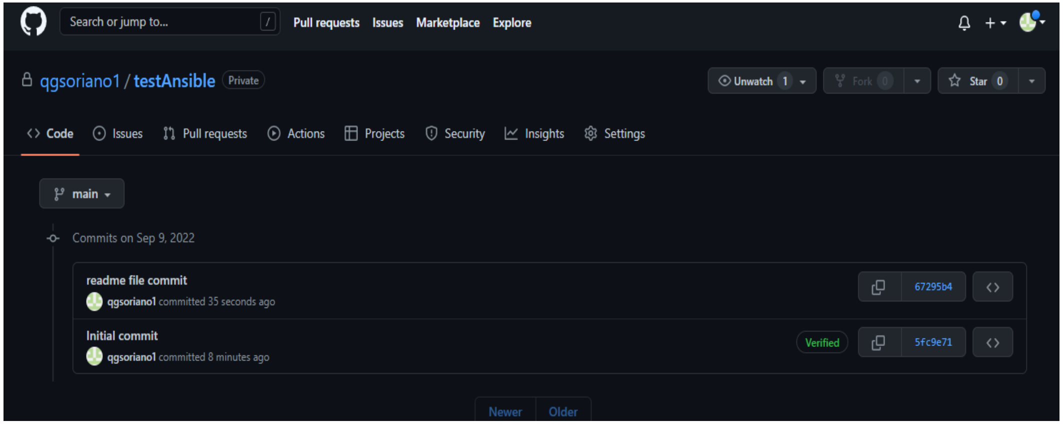
4. At this point, make sure to commit all changes to GitHub.



```
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git config --global user.
name "qgsoriano"
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git config --global user.
email qgsoriano1@tip.edu.ph
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ cat ~/.gitconfig
[user]
        name = qgsoriano
        email = qgsoriano1@tip.edu.ph
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git add README.md
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git commint -m "ansible c
ommit"
git: 'commint' is not a git command. See 'git --help'.

The most similar command is
        commit
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git commit -m "ansible co
mmit"
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$ git push origin main
Everything up-to-date
soriano2@soriano2-localmachine:~/cpe_soriano_ansible$
```



```
CPE_231_soriano  emptydir   err.txt    find.out  multitask.sh      mymessage  SampleDirect          script2.sh  shutImage.sh  std.out  update.sh
qgsoriano1@cloudshell:~$ ls README.md
ls: cannot access 'README.md': No such file or directory
qgsoriano1@cloudshell:~$ cd testAnsible
qgsoriano1@cloudshell:~/testAnsible$ ls
README.md
qgsoriano1@cloudshell:~/testAnsible$ ls README.md
README.md
qgsoriano1@cloudshell:~/testAnsible$ cat ~/.gitconfig
[user]
        name = soriano
        email = qgsoriano1@tip.edu.ph
        nam = soriano
qgsoriano1@cloudshell:~/testAnsible$ nano README.md
qgsoriano1@cloudshell:~/testAnsible$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
qgsoriano1@cloudshell:~/testAnsible$ git add README.md
qgsoriano1@cloudshell:~/testAnsible$ git commit -m "readme file commit"
[main 67295b4] readme file commit
 1 file changed, 1 insertion(+), 1 deletion(-)
qgsoriano1@cloudshell:~/testAnsible$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 255 bytes | 255.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qgsoriano1/testAnsible.git
   5fc9e71..67295b4  main -> main
qgsoriano1@cloudshell:~/testAnsible$
```



Commits on Sep 9, 2022

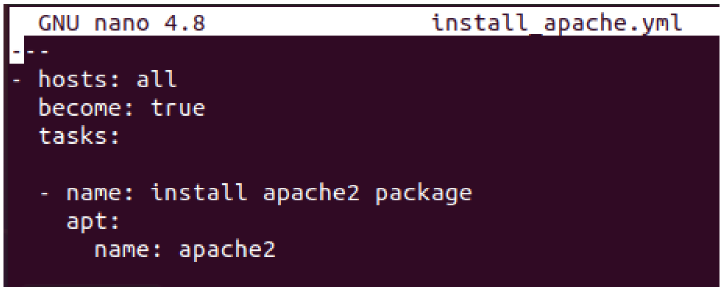
readme file commit
qgsoriano1 committed 35 seconds ago — 67295b4

Initial commit
qgsoriano1 committed 8 minutes ago — Verified — 5fc9e71

Task 2: Writing our First Playbook

1. With ad hoc commands, we can simplify the administration of remote servers. For example, we can install updates, packages, and applications, etc. However, the real strength of ansible comes from its playbooks. When we write a playbook, we can define the state that we want our servers to be in and the place or commands that ansible will carry out to bring to that state. You can use an editor to create a playbook. Before we proceed, make sure that you are in the directory of the repository that we use in the previous activities (*CPE232_yourname*). Issue the command *nano install_apache.yml*. This will create a playbook file called *install_apache.yml*. The .yml is the basic standard extension for playbook files.

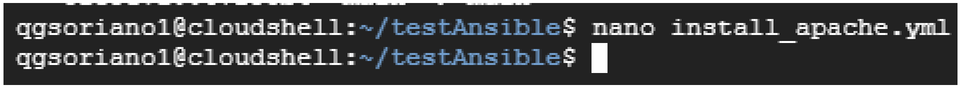
   When the editor appears, type the following:



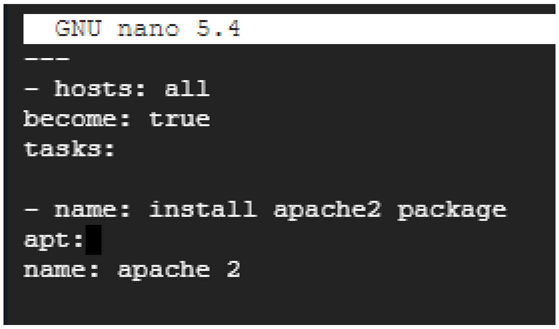
```
  GNU nano 4.8                  install_apache.yml
---
- hosts: all
  become: true
  tasks:

  - name: install apache2 package
    apt:
      name: apache2
```

   Make sure to save the file. Take note also of the alignments of the texts.

   SCREENSHOTS:

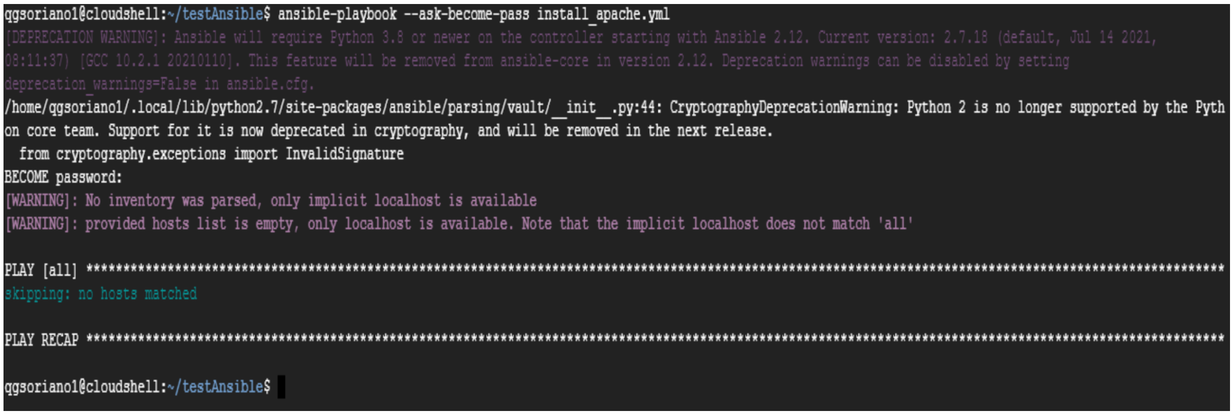


```
qgsoriano1@cloudshell:~/testAnsible$ nano install_apache.yml
qgsoriano1@cloudshell:~/testAnsible$ 
```



```
  GNU nano 5.4
---
- hosts: all
become: true
tasks:

- name: install apache2 package
apt:
name: apache 2
```

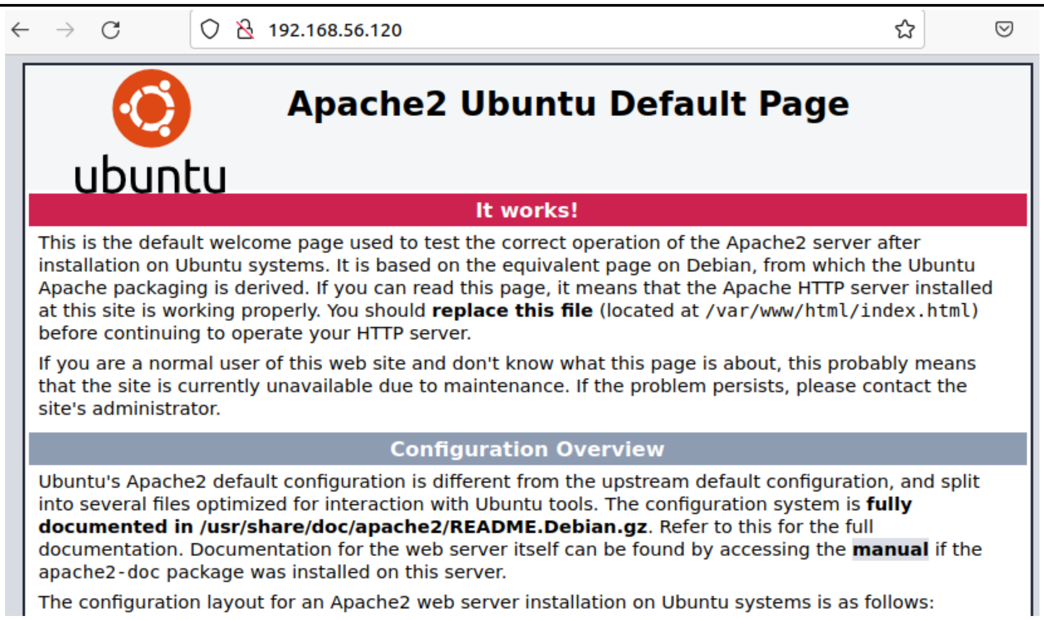
2. Run the yml file using the command: *ansible-playbook --ask-become-pass install_apache.yml.* Describe the result of this command.



```
qgsoriano1@cloudshell:~/testAnsible$ ansible-playbook --ask-become-pass install_apache.yml
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12. Current version: 2.7.18 (default, Jul 14 2021,
08:11:37) [GCC 10.2.1 20210110]. This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
/home/qgsoriano1/.local/lib/python2.7/site-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarning: Python 2 is no longer supported by the Pyth
on core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
BECOME password:
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [all] *************************************************************************************************************************************
skipping: no hosts matched

PLAY RECAP *************************************************************************************************************************************

qgsoriano1@cloudshell:~/testAnsible$ 
```

I think that the results show a successful execution of the code. I'm not that sure that it's really a success, but it looks like it.

3. To verify that apache2 was installed automatically in the remote servers, go to the web browsers on each server and type its IP address. You should see something like this.



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

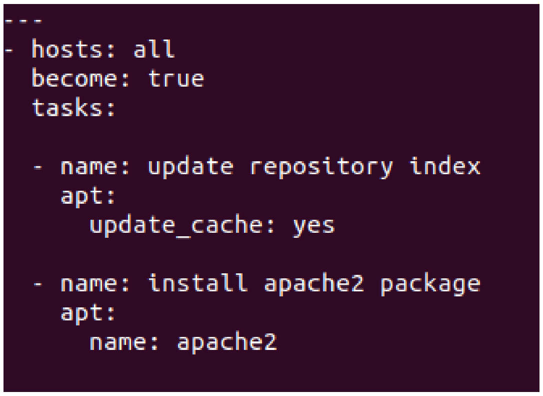
If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

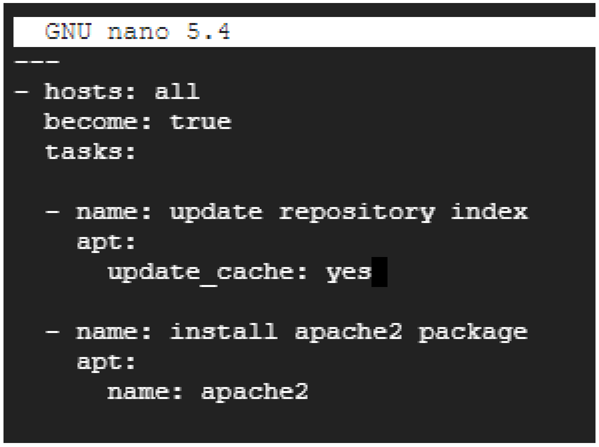
4. Try to edit the *install_apache.yml* and change the name of the package to any name that will not be recognized. What is the output?
5. This time, we are going to put additional task to our playbook. Edit the *install_apache.yml*. As you can see, we are now adding an additional command, which is the *update_cache*. This command updates existing package-indexes on a supporting distro but not upgrading installed-packages (utilities) that were being installed.



```
---
- hosts: all
  become: true
  tasks:

  - name: update repository index
    apt:
      update_cache: yes

  - name: install apache2 package
    apt:
      name: apache2
```

Save the changes to this file and exit.

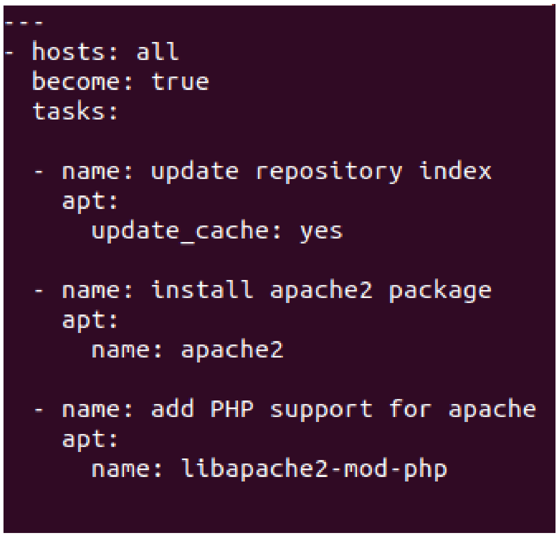
SCREENSHOT:



```
  GNU nano 5.4
---
- hosts: all
  become: true
  tasks:

  - name: update repository index
    apt:
      update_cache: yes

  - name: install apache2 package
    apt:
      name: apache2
```

6. Run the playbook and describe the output. Did the new command change anything on the remote servers?

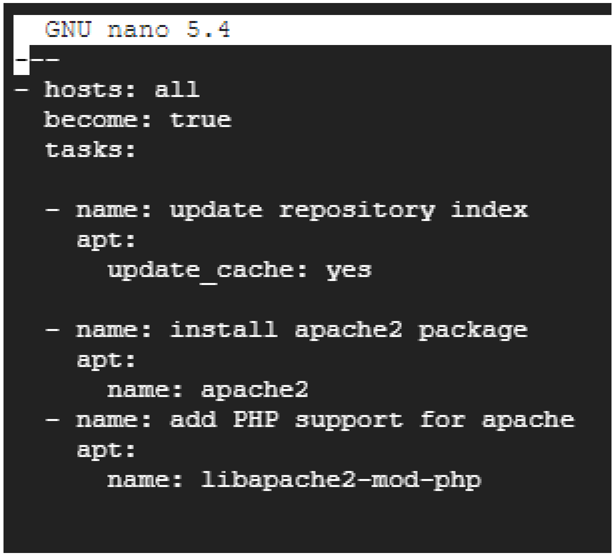
7. Edit again the *install_apache.yml*. This time, we are going to add a PHP support for the apache package we installed earlier.



```
---
- hosts: all
  become: true
  tasks:

  - name: update repository index
    apt:
      update_cache: yes

  - name: install apache2 package
    apt:
      name: apache2

  - name: add PHP support for apache
    apt:
      name: libapache2-mod-php
```

Save the changes to this file and exit.

SCREENSHOT:



```
  GNU nano 5.4
---
- hosts: all
  become: true
  tasks:

  - name: update repository index
    apt:
      update_cache: yes

  - name: install apache2 package
    apt:
      name: apache2
  - name: add PHP support for apache
    apt:
      name: libapache2-mod-php
```

8. Run the playbook and describe the output. Did the new command change anything on the remote servers?
9. Finally, make sure that we are in sync with GitHub. Provide the link of your GitHub repository.

Reflections:

Answer the following:

1. What is the importance of using a playbook?
   - **The playbook assists the team in visualizing goals, comprehending the continuous improvement paradigm, and understanding what is required to succeed. The primary steps of the workflow are described and the individual actions in those sections are outlined.**

2. Summarize what we have done on this activity.
   - **In this activity, we have done a lot of code execution. We have done a lot of installation, upgrading, and updating methods just to install the pip command and also the ansible. Most of all, we have done a lot of troubleshooting. We tried to install the command pip in the VM's Linux Ubuntu OS terminal, but it wasn't that successful because all of us have encountered a lot of problems. First was on how to install the pip in other ways, second is on how to install the ansible in many other ways. But most of these methods are not successful, just because of the**

restrictions of this “jammy” source. I tried to perform this activity on a different terminal, and I think it’s good and working, what’s not working is the remote servers, because those remote servers were inside the laboratory’s Virtual machine, and I’m not able to connect to them because I’m on a different environment and on a different PC.