

Tools Needed:

- 1. Control Node (CN) - 1
- 2. Manage Node (MN) - 1 Ubuntu
- 3. Manage Node (MN) - 1 CentOS

Procedure:

- 1. Note: You are required to create a document report of the steps you will do for this exam. All screenshots should be labeled and explained properly.
- 2. Create a repository in your GitHub account and label it as Surname_PrelimExam

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *

Repository name *

qajlsalvador

/

Salvador_PrelimExam

Salvador_PrelimExam is available.

Great repository names are short and memorable. Need inspiration? How about [bug-free-enigma](#) ?

Description (optional)

- ☒

Public

Anyone on the internet can see this repository. You choose who can commit.
- ☐

Private

You choose who can see and commit to this repository.

The creation of the repository. As the tasks suggests, I needed to name the repository with my surname in the first part combined with “PrelimExam”.

qajlsalvador / Salvador_PrelimExam

qajlsalvador

Salvador_PrelimExam

Public

Pin

Unwatch 1

Fork 0

Star 0

main 1 branch 0 tags

Go to file

Add file

Code

About

qajlsalvador Initial commit 9fb683d now 1 commit

README.md Initial commit now

README.md

Salvador_PrelimExam

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

This is confirming that I created the repository.

3. Clone your new repository in your CN.

```
salvador@Workstation:~$ git clone git@github.com:qajlsalvador/Salvador_PrelimExam.git
Cloning into 'Salvador_PrelimExam'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
salvador@Workstation:~$ ls
ansible          Documents  id_rsa.pub  Public          Templates
CPE232_AndreuSalvador Downloads  Music       Salvador_PrelimExam Videos
Desktop          id_rsa    Pictures    snap
salvador@Workstation:~$
```

Using the git clone command in the git, i was able to clone the repository that I created in the github in my home in the virtualbox. I used the code ssh from the github repository to know what is the repository that I needed to clone in my desktop (Ubuntu).

4. In your CN, create an inventory file and ansible.cfg files.

```
salvador@Workstation:~$ ls
ansible          Documents  id_rsa.pub  Public          Templates
CPE232_AndreuSalvador Downloads  Music       Salvador_PrelimExam Videos
Desktop          id_rsa    Pictures    snap
salvador@Workstation:~$ cd Salvador_PrelimExam
salvador@Workstation:~/Salvador_PrelimExam$
```

Here I use the cd command in order to go to the directory that I cloned in github.

```
salvador@Workstation: ~/Salvador_PrelimExam
GNU nano 6.2 inventory
192.168.0.145
192.168.0.146
```

Creating the inventory where I put the ip address of the servers I will be using.

```
GNU nano 6.2 ansible.cfg
[defaultls]
inventory = /home/salvador/Salvador_PrelimExam/inventory
```

Creating the .cfg file for my ansible to know where to access those ip address.

```
salvador@Workstation: ~/Salvador_PrelimExam$ sudo nano ansible.cfg
salvador@Workstation: ~/Salvador_PrelimExam$ sudo nano inventory
salvador@Workstation: ~/Salvador_PrelimExam$ sudo nano inventory
salvador@Workstation: ~/Salvador_PrelimExam$ ls
ansible.cfg  inventory  README.md
salvador@Workstation: ~/Salvador_PrelimExam$
```

Confirming that I have created the necessary initial files in order to play the playbook.

5. Create an Ansible playbook that does the following with an input of a config.yaml file for both Manage Nodes

```
salvador@Workstation: ~/Salvador_PrelimExam
GNU nano 6.2 config.yaml *
---
- hosts: all
  become: true
  vars:
    uname: AndreuSalvador
  tasks:
```

Creating the yaml file. Put the necessary commands in order to start the yaml file. Hosts has all for it to run every hosts in my inventory.

- Installs the latest python3 and pip3

Code:

```
- name: Install and python3 in Ubuntu and CentOS
package:
  name: python3
  state: latest

- name: install pip in Ubuntu and CentOS
package:
  name: python3-pip
  state: latest
```

The code for the installation of the python3 and the pip3. I used the package command instead of dnf and apt so that it will automatically identify if the ip address of that server is Ubuntu which will put the appropriate package like apt and dnf for the CentOS. In this way I can lessen my lines of code.

- use pip3 as default pip
- use python3 as default python

```
- name: Make pip3 as the default pip
shell:
  echo 'alias pip=/home/trilift/.local/bin/pip3 >> ~/.bashrc'
  echo 'alias python=/usr/bin/python3 >> ~/.bashrc'
```

Code for making the pip3 and python3 as the default pip and python for my servers to use.

- Install Java open-jdk

```
- name: Install openjdk on Ubuntu
apt:
  name: -openjdk-11-jdk
  state: latest
when: ansible_distribution == "Ubuntu"

- name: Install openjdk on CentOS
dnf:
  name: java-11-openjdk-devel
  state: latest
when: ansible_distribution == "CentOS"
```

I've had problems running the jdk package as it has different commands for different OS, so I had to separate the two in order to work. Using apt for the Ubuntu server and dnf for the CentOS as it is the appropriate package command to use.

- Create Motd containing the text defined by a variable defined in config.yaml file and if there is no variable input the default motd is "Ansible Managed node by (your user name)"

```
- name: making an motd
  debug:
    msg: " Ansible Managed node by {{uname}} "
```

Created the motd or message of the day with the specific string given by the instructions.

- Create a user with a variable defined in config.yaml

```
- name: Creating a user
  ansible.builtin.user:
    name: "{{ uname }}"
    createhome: yes
    home: "/home/{{ uname }}"
```

Creating a default user in my home/desktop.

Run:

Ubuntu Server:

```
salvador@Workstation: ~/Salvador_PrelimExam
GNU nano 6.2 inventory *
192.168.0.147
#192.168.0.145
```

```
salvador@Workstation:~/Salvador_PrelimExam$ ansible-playbook -i inventory --ask-become-pass config.yaml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [192.168.0.147]

TASK [Install and python3 in Ubuntu and CentOS] *****
ok: [192.168.0.147]

TASK [install pip in Ubuntu and CentOS] *****
ok: [192.168.0.147]

TASK [Install openjdk on Ubuntu] *****
ok: [192.168.0.147]

TASK [Install openjdk on CentOS] *****
skipping: [192.168.0.147]

TASK [Make pip3 as the default pip] *****
changed: [192.168.0.147]

TASK [making an motd] *****
ok: [192.168.0.147] => {
  "msg": " Ansible Managed node by AndreuSalvador "
}

TASK [Creating a user] *****
changed: [192.168.0.147]

PLAY RECAP *****
192.168.0.147      : ok=7    changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

I had to comment out first the other ip address for it to now give me an error when running because my pc seems to not handle opening 2 servers at the same time. Ok indicates that the supposedly packages are already installed in the serve, changed indicates that an installation have occurred which in my case in this Ubuntu server, the pip3 and python 3 wasn't the default python and pip as well as haven't created a user under that name yet.

CentOS server:

```
salvador@Workstation: ~/Salvador_PrelimExam
GNU nano 6.2 inventory *
#192.168.0.147
192.168.0.145
```

```
salvador@Workstation:~/Salvador_PrelimExam$ ansible-playbook -i inventory --ask-become-pass config.yaml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [192.168.0.145]

TASK [Install and python3 in Ubuntu and CentOS] *****
ok: [192.168.0.145]

TASK [install pip in Ubuntu and CentOS] *****
ok: [192.168.0.145]

TASK [Install openjdk on Ubuntu] *****
skipping: [192.168.0.145]

TASK [Install openjdk on CentOS] *****
ok: [192.168.0.145]

TASK [Make pip3 as the default pip] *****
changed: [192.168.0.145]

TASK [making an motd] *****
ok: [192.168.0.145] => {
  "msg": " Ansible Managed node by AndreuSalvador "
}

TASK [Creating a user] *****
changed: [192.168.0.145]

PLAY RECAP *****
192.168.0.145      : ok=7    changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

The second server that I played the ansible playbook was the CentOS server. Again, I comment out the other server which is now the Ubuntu server, in order to not show any error that the ip address I was trying to access is inaccessible. Just like in the previous server the ok indicates that it was already installed. The case for both servers are the same, the pip3 and python3 wasn't the default pip and python, as well as haven't created user under that name indicated by my code. Skipped means that it is not for the right server.

Checking the installation:

Ubuntu:

Python3:

```
Try 'python -h' for more information.
salvador@Server2:~$ python3 --version
Python 3.10.12
salvador@Server2:~$ _
```

Java:

```
salvador@Server2:~$ java -version
openjdk version "11.0.20.1" 2023-08-24
OpenJDK Runtime Environment (build 11.0.20.1+1-post-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.20.1+1-post-Ubuntu-0ubuntu122.04, mixed mode, sharing)
salvador@Server2:~$
```

CentOS:

Python3:

```
* updates: repo.shinjiru.com
Package python3-3.6.8-19.el7_9.x86_64 already installed and latest version
Nothing to do
[salvador@salvadorcentos ~]$ python --ver
```

Java:

```
* updates: repo.shinjiru.com
Package 1:java-11-openjdk-devel-11.0.20.0.8-1.el7_9.x86_64 already installed and latest version
Nothing to do
[salvador@salvadorcentos ~]$ java -version
openjdk version "11.0.20.1" 2023-08-24
```


5. PUSH and COMMIT your PrelimExam in your GitHub repo

```
salvador@Workstation:~/Salvador_PrelimExam$ git add config.yaml inventory
salvador@Workstation:~/Salvador_PrelimExam$ git commit -m "Prelim Exam Finished"
[main 130b720] Prelim Exam Finished
 2 files changed, 46 insertions(+)
 create mode 100644 config.yaml
 create mode 100644 inventory
salvador@Workstation:~/Salvador_PrelimExam$
```

Adding the two files in the github repository which are the yml file and the inventory file.

Committing so that the changes will be noticed when I access my github repository

```
create mode 100644 inventory
salvador@Workstation:~/Salvador_PrelimExam$ git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 769 bytes | 769.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:qajlsalvador/Salvador_PrelimExam.git
 9fb683d..130b720  main -> main
```

Confirming that the files were added and pushes it in the repository

Salvador_PrelimExam

Public

Pin

Unwatch 1

main

1 branch

0 tags

Go to file

Add file

Code

qajlsalvador Prelim Exam Finished

130b720 2 minutes ago 2 commits

README.md	Initial commit	3 days ago
config.yaml	Prelim Exam Finished	2 minutes ago
inventory	Prelim Exam Finished	2 minutes ago

README.md

Salvador_PrelimExam

Showing that the files we're successfully added to the repository

6. For your prelim exam to be counted, please paste your repository link here.

https://github.com/qajlsalvador/Salvador_PrelimExam.git