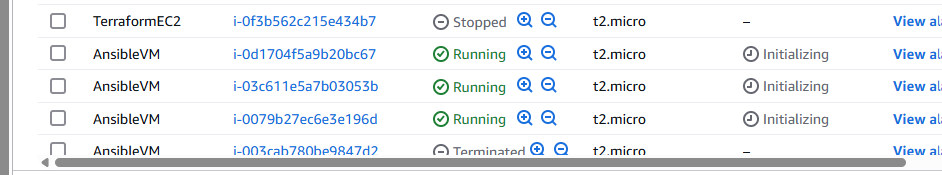
Ansible part II

Configuration management tool

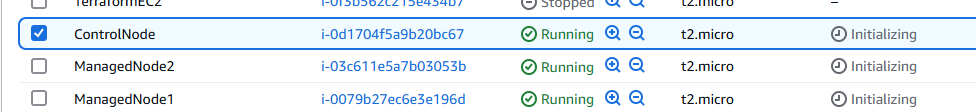
Setup: Setting up to work with Ansible

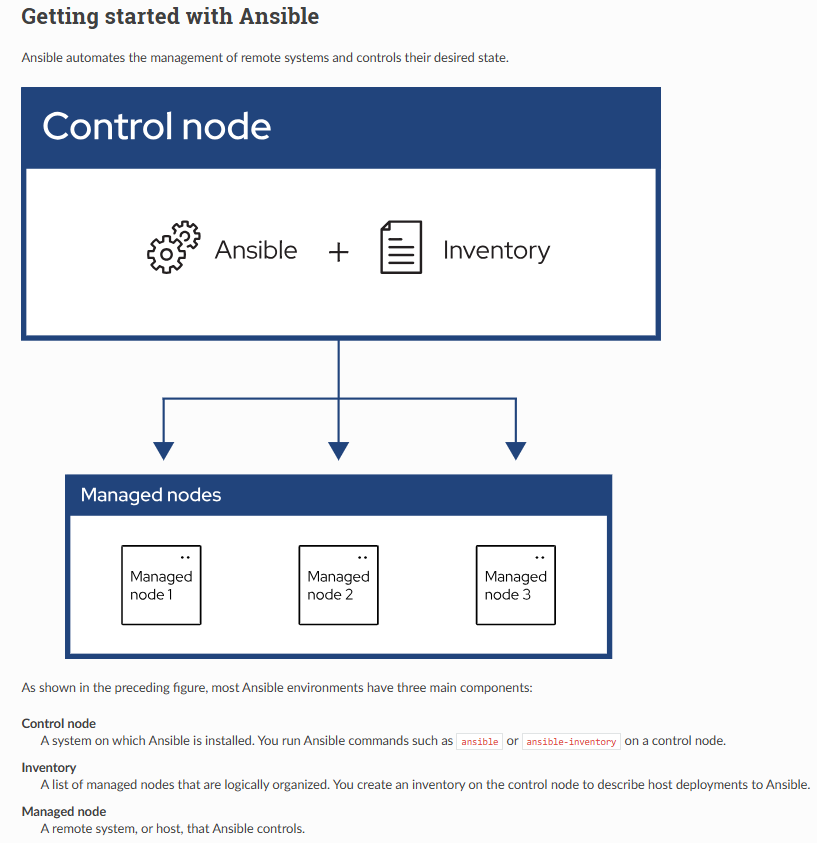
1. Create 3 Linux VMs in AWS
2. 2 VMs as Managed nodes
3. 1 VM as Control node

I have created 3 machines:



Rename VMs





Setup User and Configure user in Sudoers file and update

SSHD config file in all 3 VMs

1. Create User (All 3 VMs)

sudo useradd ansible

sudo passwd ansible

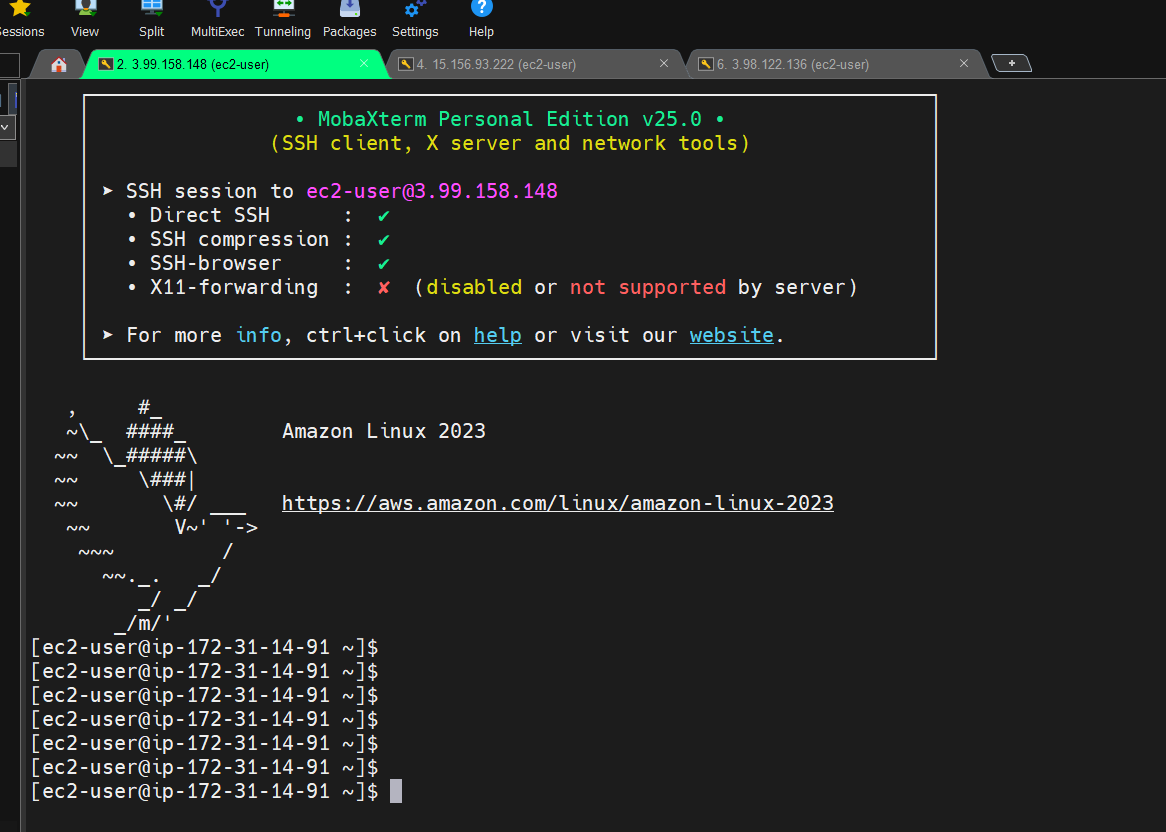
1. Configure user in sudoers file
   * 1. sudo visudo
2. ansible ALL = (ALL), NoPasswd: ALL
3. Update SSHD config file: sudo vi /etc/ssh/sshd\_config
4. Install Ansible in ControlNode: Switch to Ansible user sudo su ansible
   * 1. cd ~
5. Install Python
   * 1. sudo yum install python3 -y
6. To check Python version
   * 1. python3 --version
7. Install pip (Python package manager)
   * 1. sudo yum -y install python3-pip
8. Install Ansible using Python PIP

pip3 install ansible --user

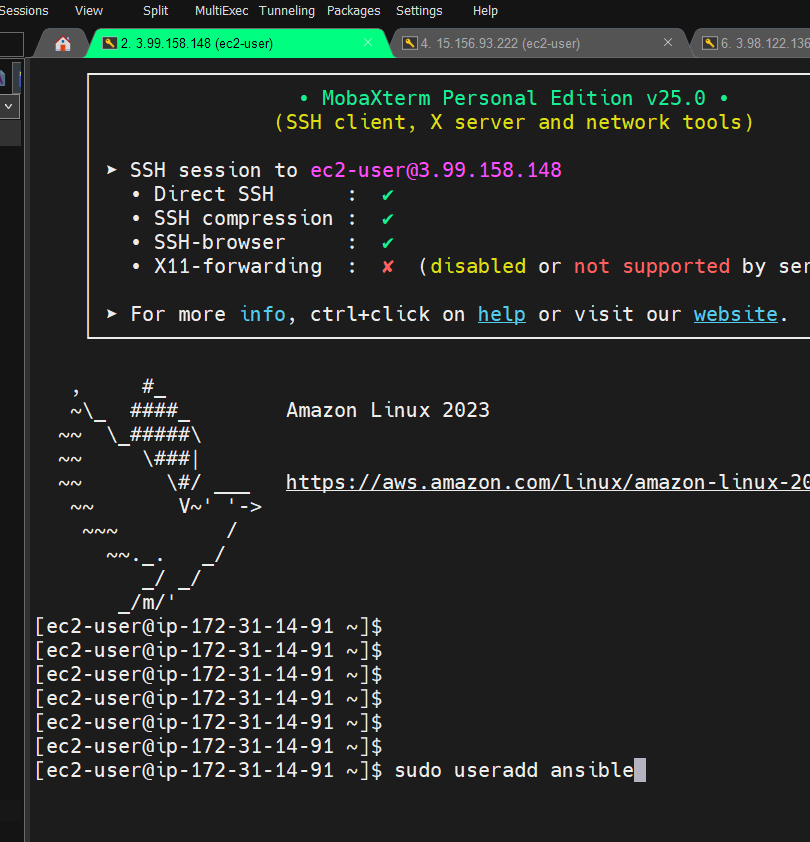
1. Verify Ansible version
   * 1. ansible --version
2. Create Ansible folder under /etc
   * 1. sudo mkdir /etc/ansible
3. Generate SSH key in our Control Node & copy SSH key into Managed Nodes
4. Switch to Ansible user
   1. sudo su ansible
5. Generate SSH key
   1. ssh-keygen (just push enter-enter-enter) don’t write anything
6. Copy it to Managed Nodes as Ansible user
   * 1. ssh-copy-id ansible@<ManagedNode-PrivateIP>
7. Update Host inventory in Ansible server
   * 1. sudo vi /etc/ansible/hosts
8. Test connectivity
   * 1. ansible webservers -m ping

Create user ansible in all 3 VMs: sudo useradd ansible

Green tab is the control node others are managed nodes

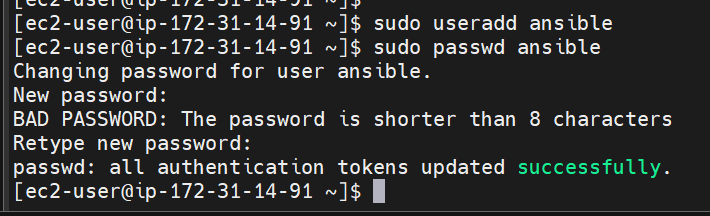


In the Control Node



[ec2-user@ip-172-31-14-91 ~]$ sudo useradd ansible

[ec2-user@ip-172-31-14-91 ~]$ sudo passwd ansible



Do the same thing in Managed User also

[ec2-user@ip-172-31-5-77 ~]$ sudo passwd ansible

Changing password for user ansible.

New password:

BAD PASSWORD: The password is shorter than 8 characters

Retype new password:

passwd: all authentication tokens updated successfully.

1. Create User (All 3 VMs)

sudo useradd ansible

sudo passwd ansible

1. Configure user in sudoers file
   * 1. sudo visudo
2. ansible ALL = (ALL), NoPasswd: ALL
3. Update SSHD config file: sudo vi /etc/ssh/sshd\_config

Install Ansible in Control Node

Open sudoers file:

[ec2-user@ip-172-31-5-77 ~]$ sudo visudo

Scroll down and say

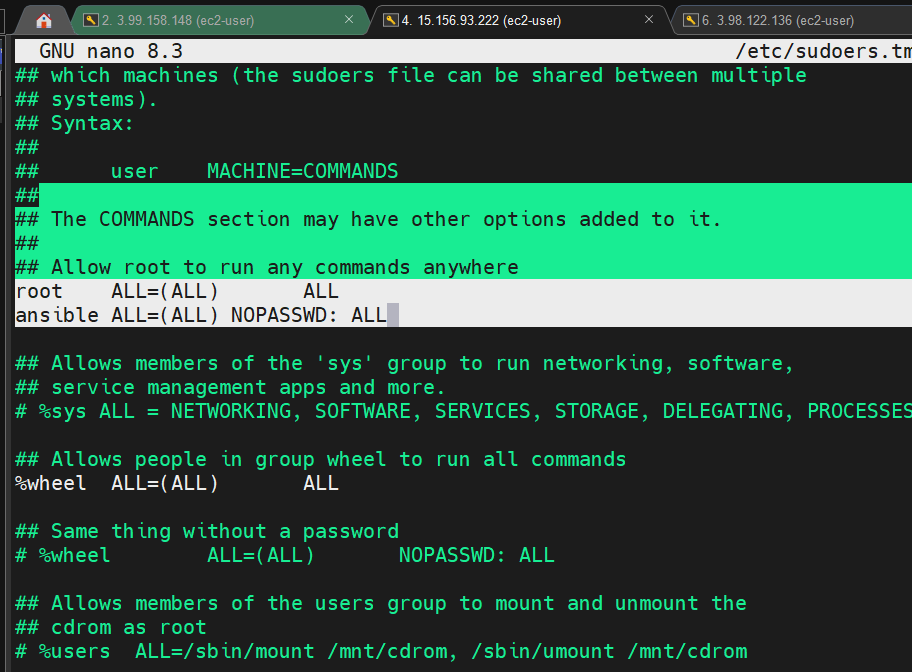
## The COMMANDS section may have other options added to it.

##

## Allow root to run any commands anywhere

root ALL=(ALL) ALL

ansible ALL=(ALL) NOPASSWD: ALL



Ctrl + X --> push Y, enter

DO the same things in all 3 VMs

ansible ALL=(ALL) NOPASSWD: ALL

Next:

[ec2-user@ip-172-31-14-91 ~]$ sudo vi /etc/ssh/sshd\_config

Change PermitEmptyPasswords to yes

# Explicitly disable PasswordAuthentication. By presetting it, we

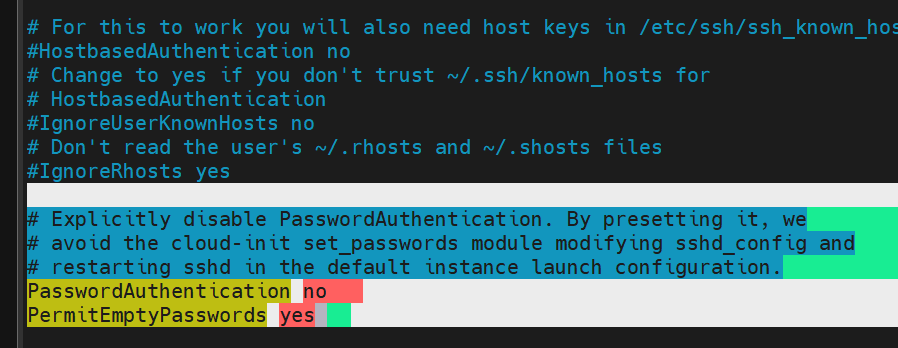
# avoid the cloud-init set\_passwords module modifying sshd\_config and

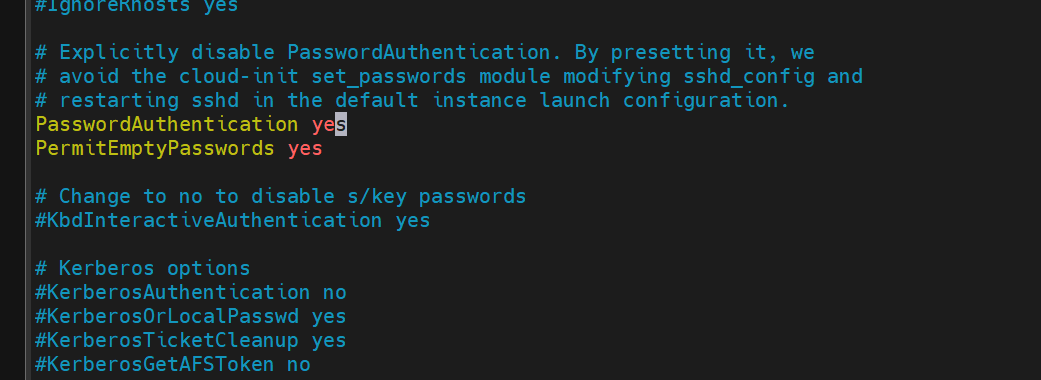
# restarting sshd in the default instance launch configuration.

PasswordAuthentication yes

PermitEmptyPasswords yes

PubkeyAuthentication yes





Same step please do in other machines as well

Restart sshd service in all 3 VMs

[ec2-user@ip-172-31-14-91 ~]$ sudo service sshd restart

Redirecting to /bin/systemctl restart sshd.service

Install Ansible in Control Node

[ec2-user@ip-172-31-14-91 ~]$ sudo yum install python3 -y

[ec2-user@ip-172-31-14-91 ~]$ sudo yum install python3 -y

Amazon Linux 2023 Kernel Livepatch repository 140 kB/s | 15 kB 00:00

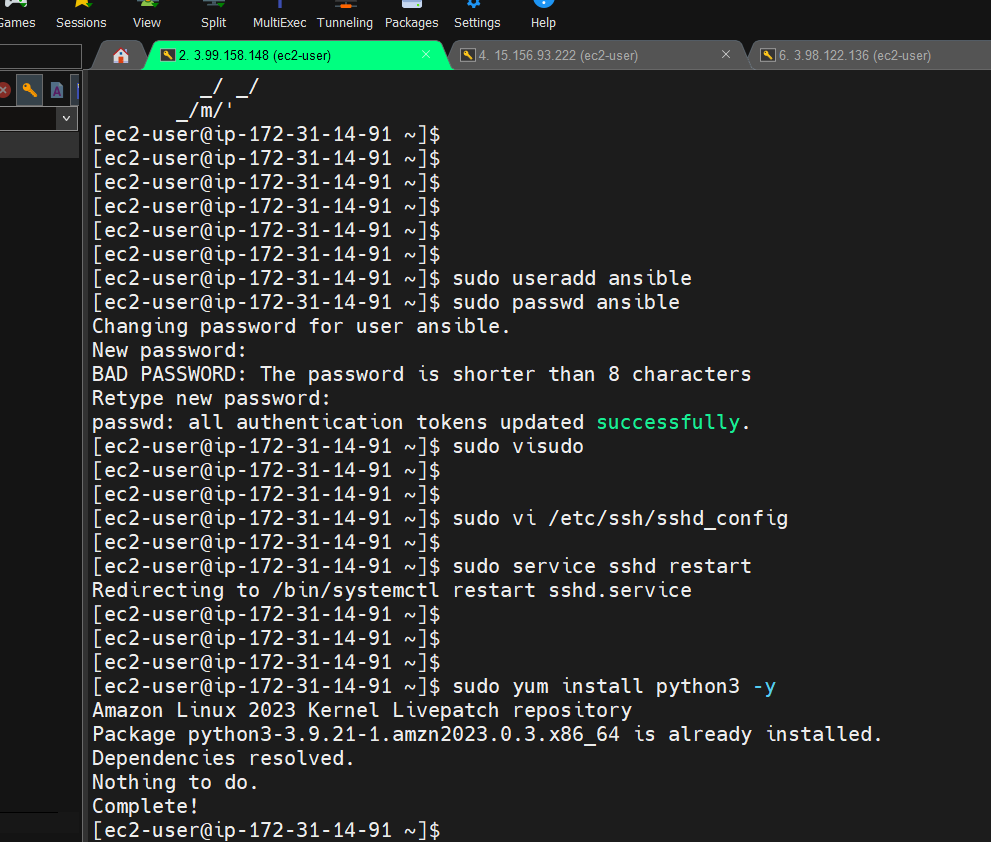
Package python3-3.9.21-1.amzn2023.0.3.x86\_64 is already installed.

Dependencies resolved.

Nothing to do.

Complete!

Installed Python for Ansible only in Control Node



[ec2-user@ip-172-31-14-91 ~]$ sudo yum -y install python3-pip

[ec2-user@ip-172-31-14-91 ~]$ pip3 install ansible --user

[ec2-user@ip-172-31-14-91 ~]$ ansible --version

ansible [core 2.15.13]

config file = None

configured module search path = ['/home/ec2-user/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']

ansible python module location = /home/ec2-user/.local/lib/python3.9/site-packages/ansible

ansible collection location = /home/ec2-user/.ansible/collections:/usr/share/ansible/collections

executable location = /home/ec2-user/.local/bin/ansible

python version = 3.9.21 (main, Mar 19 2025, 00:00:00) [GCC 11.5.0 20240719 (Red Hat 11.5.0-5)] (/usr/bin/python3)

jinja version = 3.1.6

libyaml = True

[ec2-user@ip-172-31-14-91 ~]$ sudo mkdir /etc/ansible

[ec2-user@ip-172-31-14-91 ~]$ sudo su ansible

[ansible@ip-172-31-14-91 ec2-user]$ ssh-keygen

Generating public/private rsa key pair.

Enter file in which to save the key (/home/ansible/.ssh/id\_rsa):

Created directory '/home/ansible/.ssh'.

Enter passphrase (empty for no passphrase):

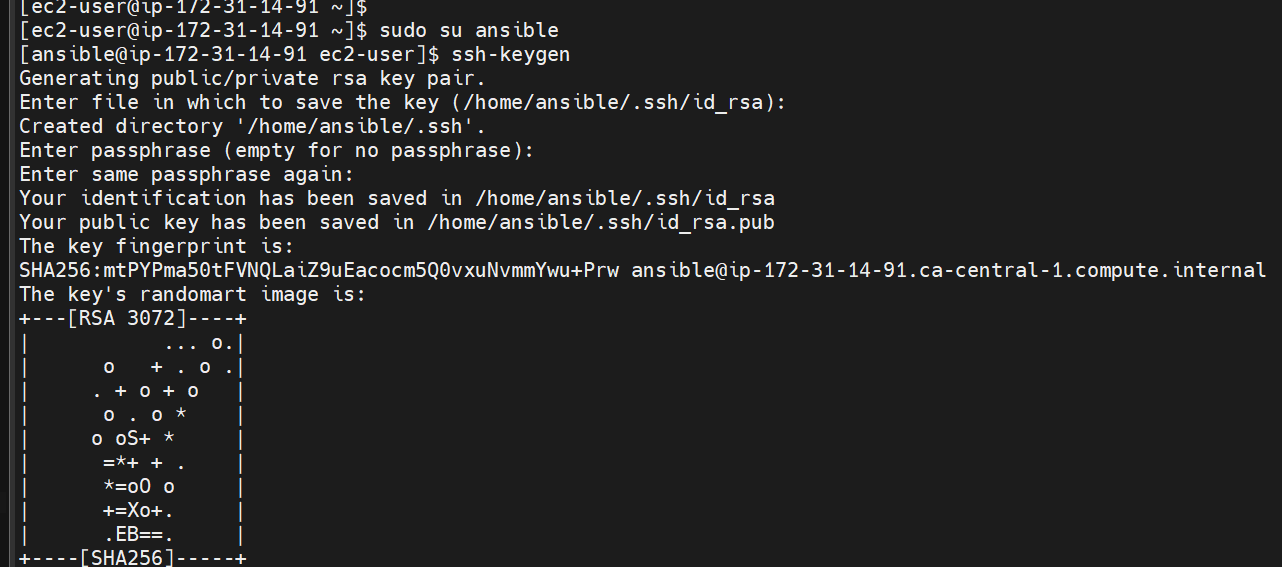
Enter same passphrase again:

Your identification has been saved in /home/ansible/.ssh/id\_rsa

Your public key has been saved in /home/ansible/.ssh/id\_rsa.pub

The key fingerprint is:

SHA256:mtPYPma50tFVNQLaiZ9uEacocm5Q0vxuNvmmYwu+Prw ansible@ip-172-31-14-91.ca-central-1.compute.internal



ssh-copy-id ansible@<ManagedNode-PrivateIP>

ManagedNode1 PrivateIP: 172.31.5.77

ManagedNode2 PrivateIP: 172.31.12.31

Run this in the Control Node

ssh-copy-id [ansible@172.31.5.77](mailto:ansible@172.31.5.77)

ssh-copy-id [ansible@172.31.12.31](mailto:ansible@172.31.12.31)

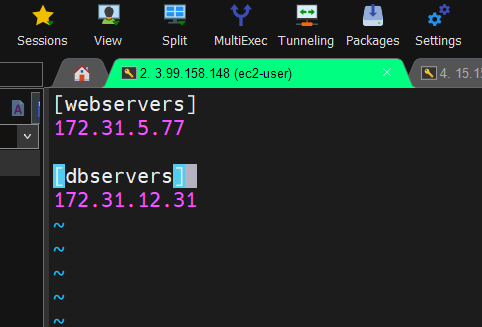
What’s Host inventory? All Managed Nodes

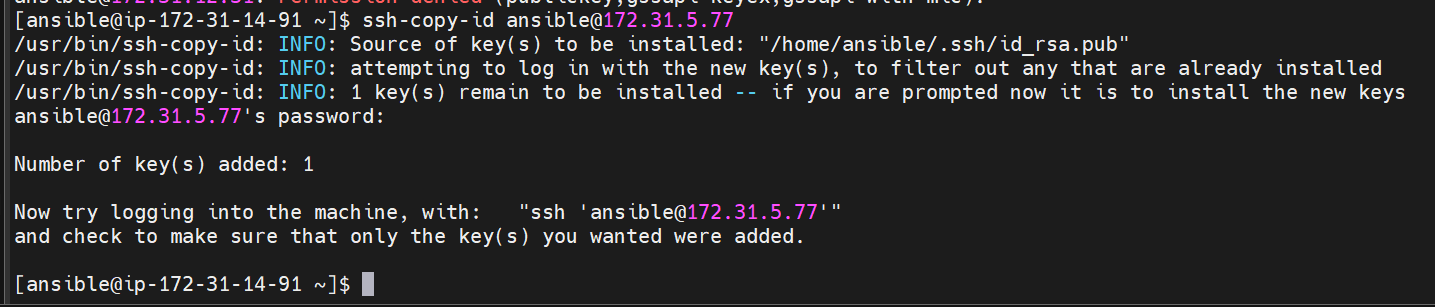
Update Host inventory in Ansible server to add Managed Node Server details

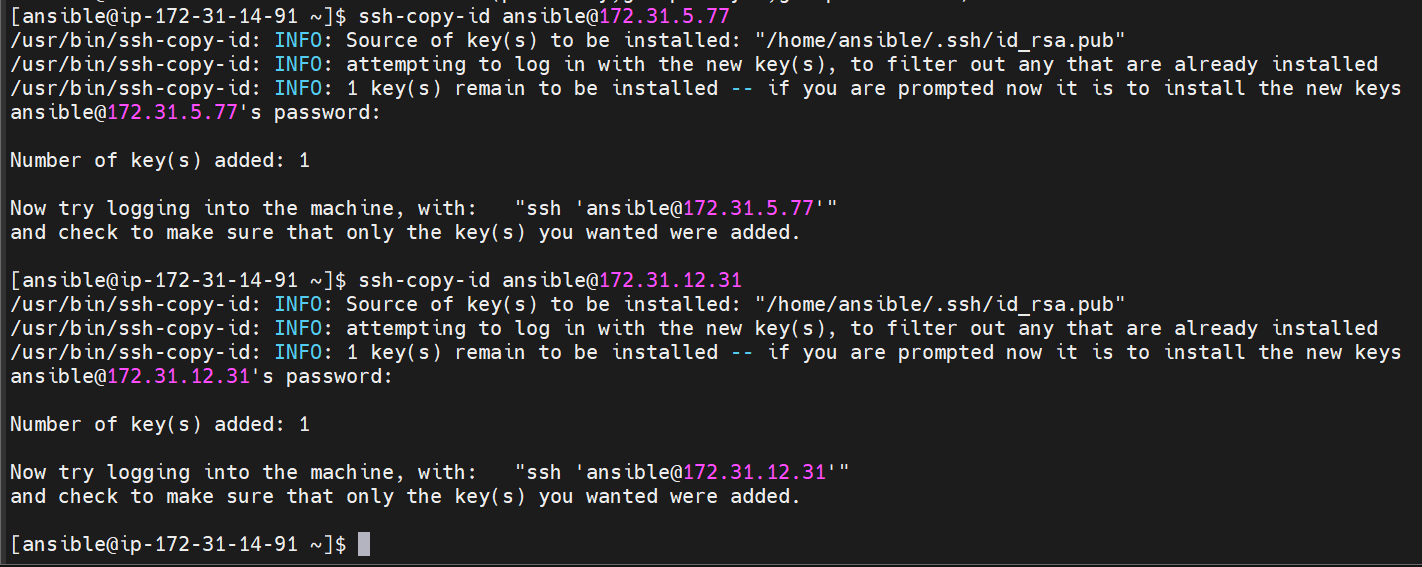
sudo vi /etc/ansible/hosts

In Control Node

[ansible@ip-172-31-14-91 ec2-user]$ sudo vi /etc/ansible/hosts







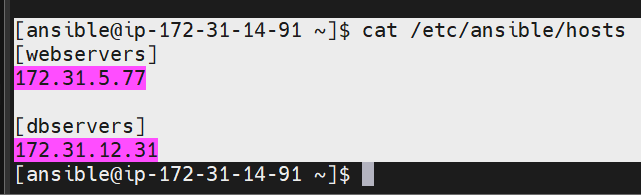
[ansible@ip-172-31-14-91 ~]$ cat /etc/ansible/hosts

[webservers]

172.31.5.77

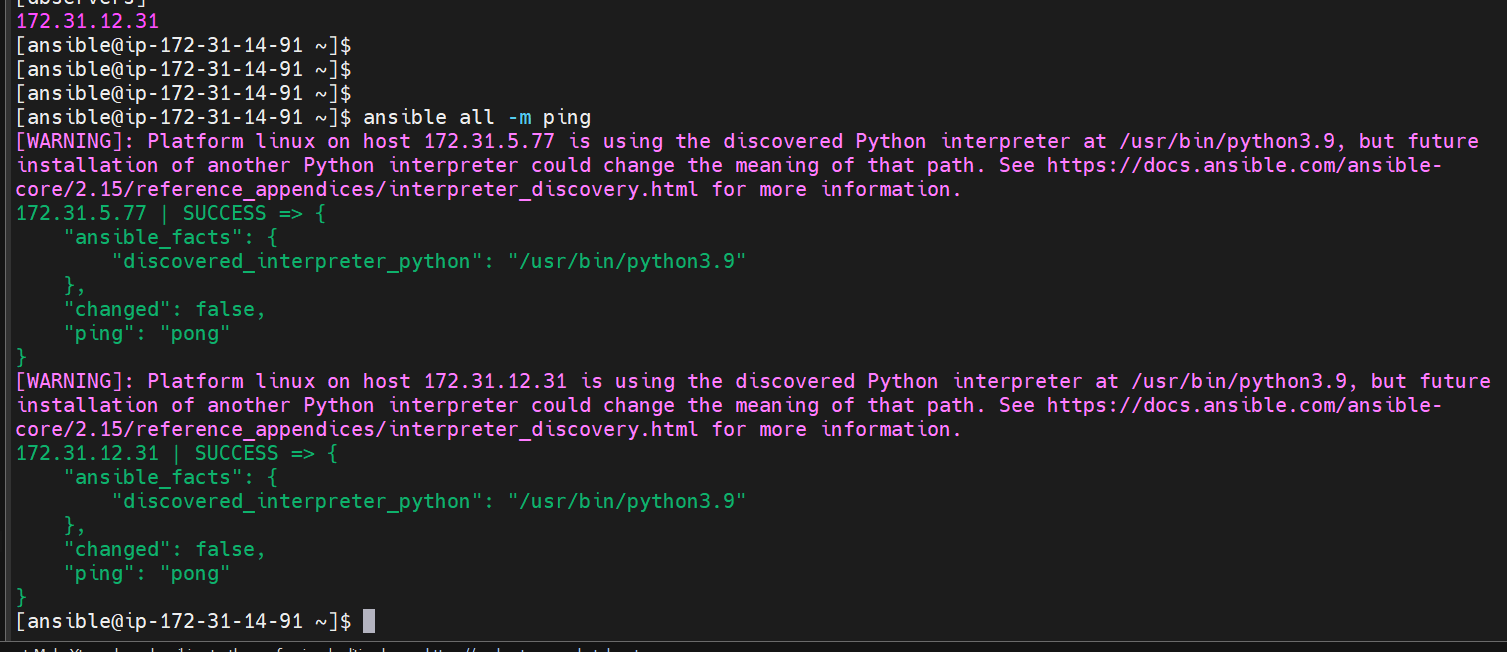
[dbservers]

172.31.12.31

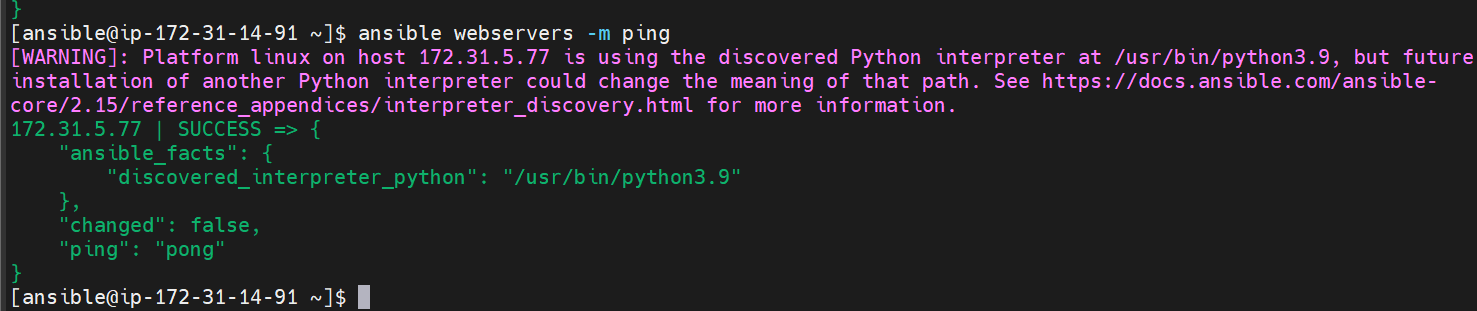


We should get SUCCESS

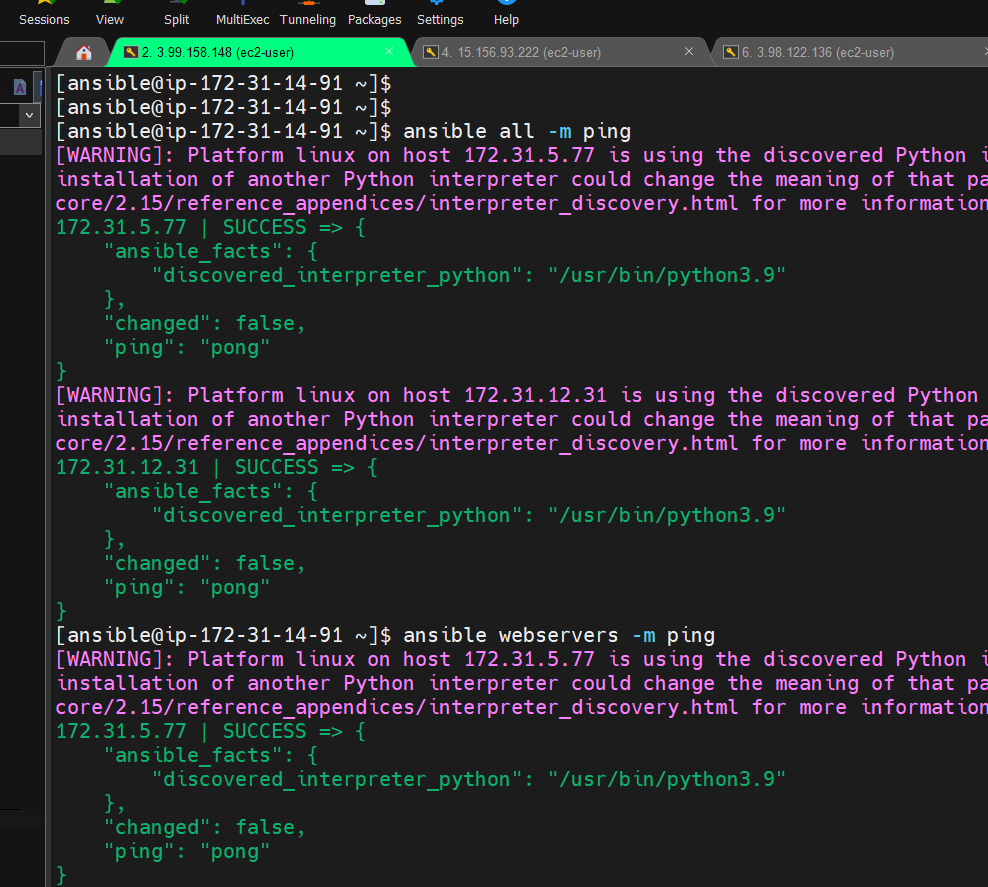
[ansible@ip-172-31-14-91 ~]$ ansible all -m ping



[ansible@ip-172-31-14-91 ~]$ ansible webservers -m ping



Everything from Control Node



Ansible Ad-Hoc commands

Ansible [all/group-name/host-name/ip] -m <module> -a <args>

ansible all -m ping

We have many modules in ansible to perform configuration management

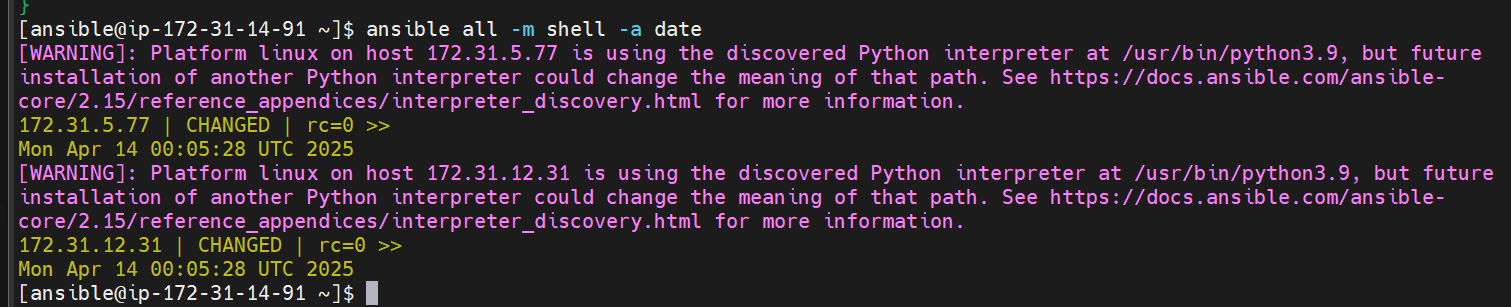
ping, shell, yum, copy, service

***ping module --> $ ansible all -m ping, ansible webservers -m ping***

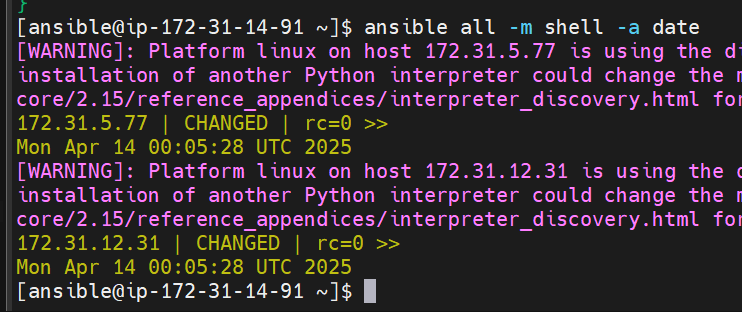
***shell moule --> $ ansible all -m shell -a date, $ ansible all -m shell -a uptime***

***yum module --> $ ansible webservers -b -m yum -a “name=git”***

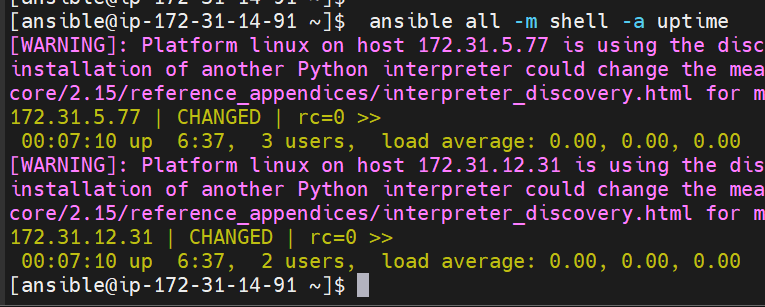
Get date from both machines



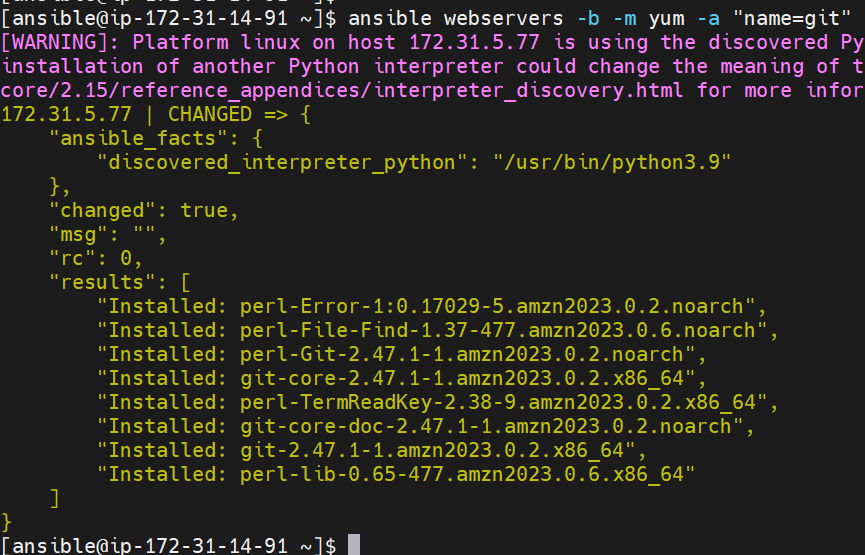
**Manage machines remotely**



ansible all -m shell -a uptime



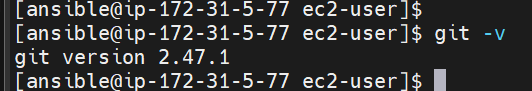
[ansible@ip-172-31-14-91 ~]$ ansible webservers -b -m yum -a "name=git"



Installed git-core-2.47.1 in the Amazon machine

Now I go into Managed VM1

Git is already installed remotely from Control Node



Ansible Playbooks

Playbook is a YAML file, which contains one or more tasks

Using Playbook we define what tasks to be performed and where to be performed

Example YML format

Student:

Id: 5

Name: Abc

Gender: Male

Hobbies:

- Cricket

- Chess

- VideoGame

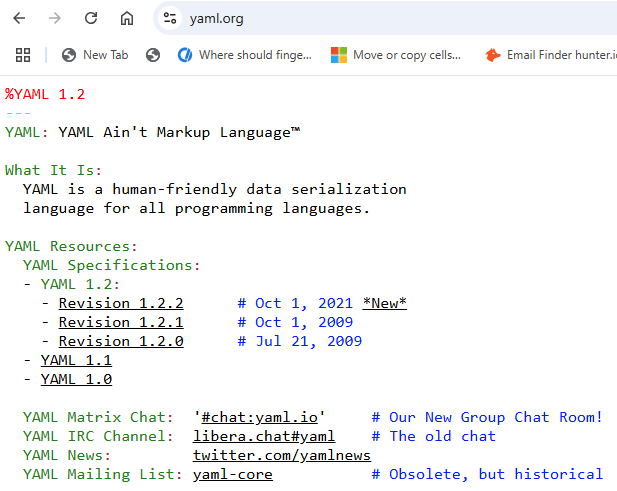
Address:

City: Cincinatti

Country: USA

Note: Ident spaces are the most important in yml file

Website: <https://yaml.org/> shows the syntax



For yaml syntax validation: <https://www.yamllint.com/>

'''

Student:

Id: 5

Name: Abc

Gender: Male

Hobbies:

- Cricket

- Chess

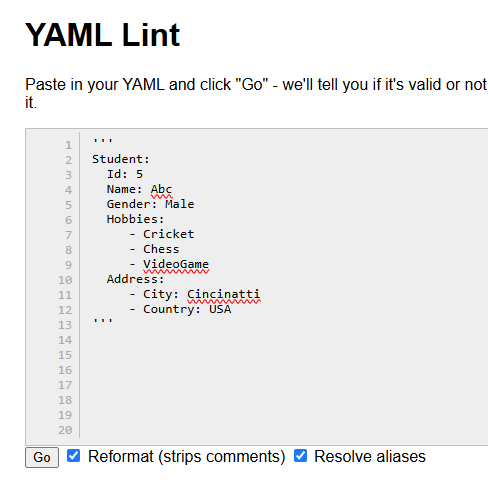
- VideoGame

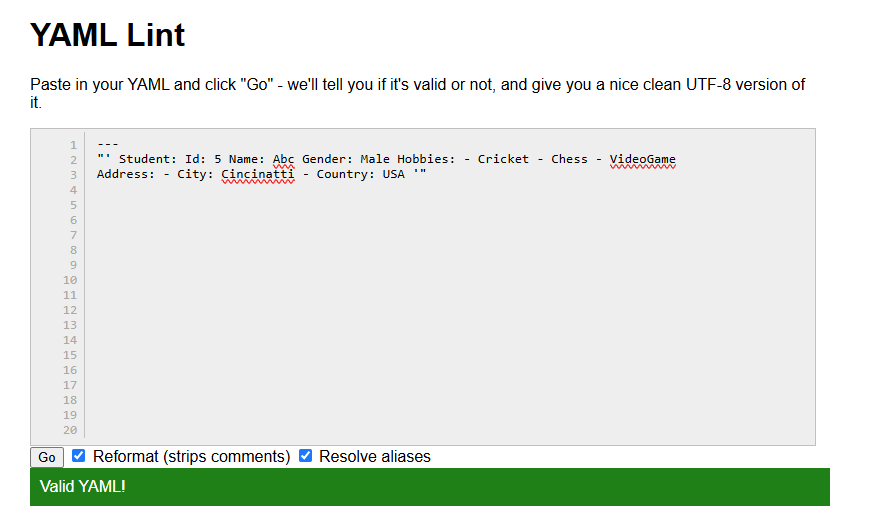
Address:

- City: Cincinatti

- Country: USA

'''





Hit Go

Yaml stats with --- and ends with …

Writing Playbooks

3 major sections: Host section, Variable section, Task section

Host section: Represents Target machines to execute tasks. This configuration depends on Ansible inventory file

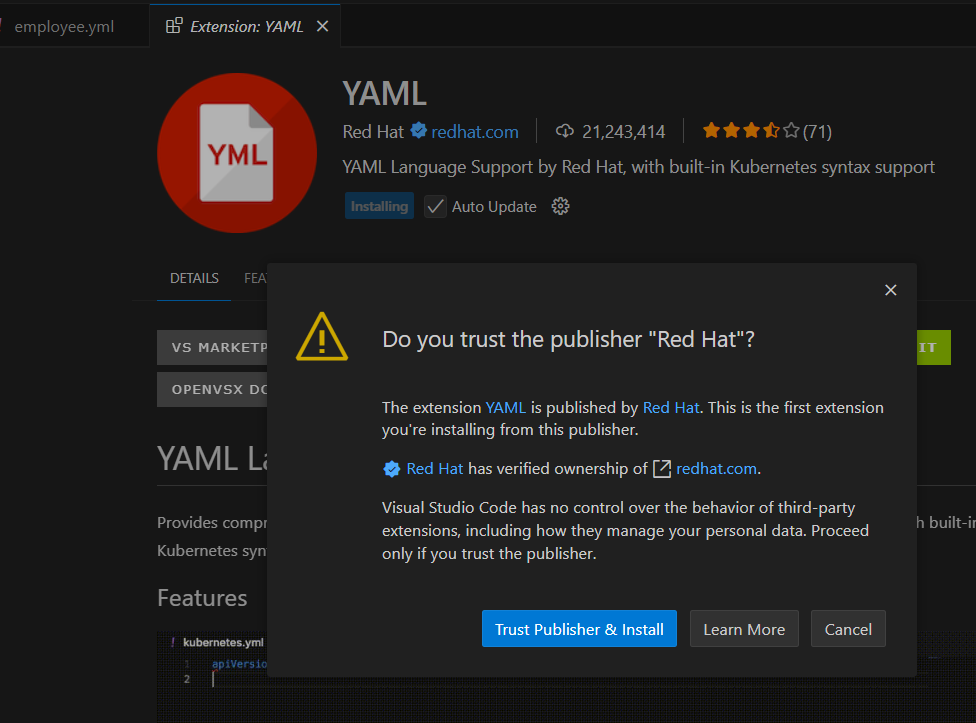
Variable section: used to declare variables in Playbook execution

Task section: defines what operations or tasks we want to perform using Ansible

$ ansible-playbook <playbook-yml-filename>

Open VSCode

Install



---

employee:

id: 01

name: Abc

job:

experience: 5 years

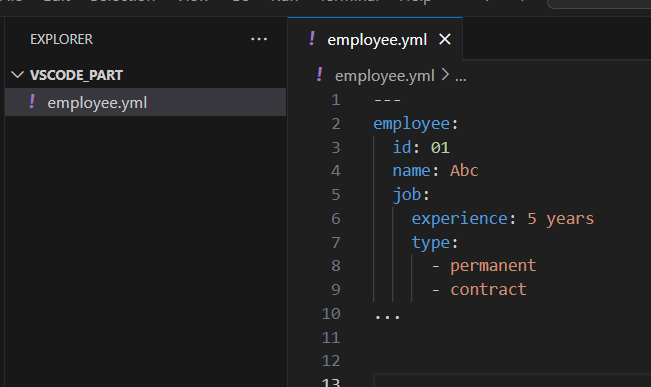
type:

- permanent

- contract

...

Create employee.yml



---

employee:

id: 01

name: Abc

job:

experience: 5 years

type:

- permanent

- contract

student:

id: 5

name: Xyz

skills:

- edit videos

- graphic design

…

---

- hosts: all

tasks:

- name: Ping all the managed nodes

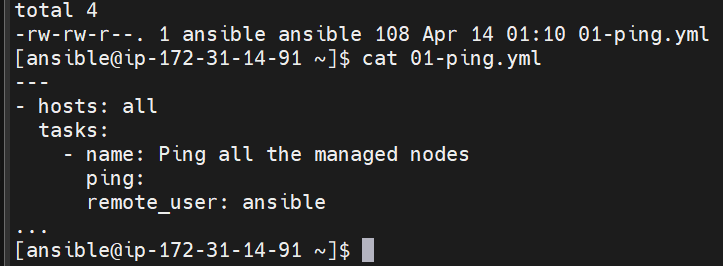
ping:

remote\_user: ansible

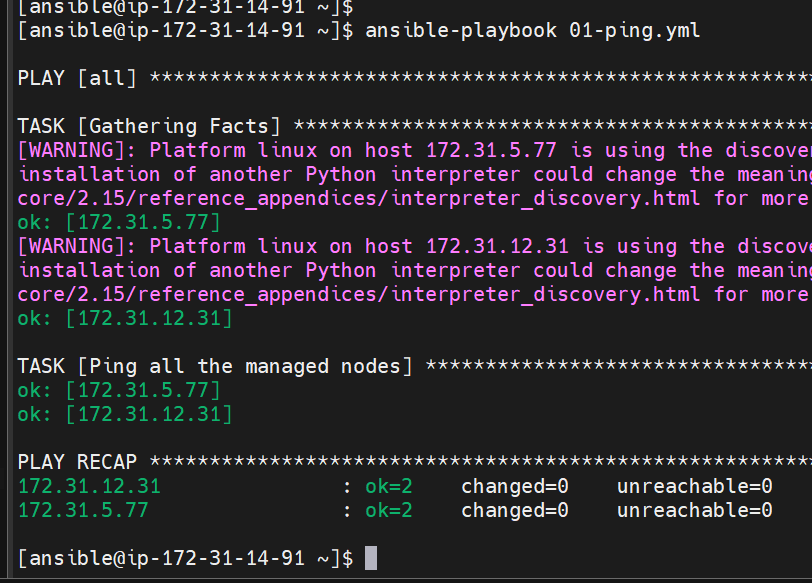
...

Go to Control Node VM

[ansible@ip-172-31-14-91 ~]$ vi 01-ping.yml



[ansible@ip-172-31-14-91 ~]$ ansible-playbook 01-ping.yml



[ansible@ip-172-31-14-91 ~]$ ansible-playbook 01-ping.yml --syntax-check

playbook: 01-ping.yml