Tasneem Sherif Tawfik Ali Yousif
AST-DevOps Engineer_ONL3_SWD1_G2

🧪 Ansible Assignment: Basic Setup and Configuration

## 🎯 Objective

Practice Ansible fundamentals including:

- SSH setup

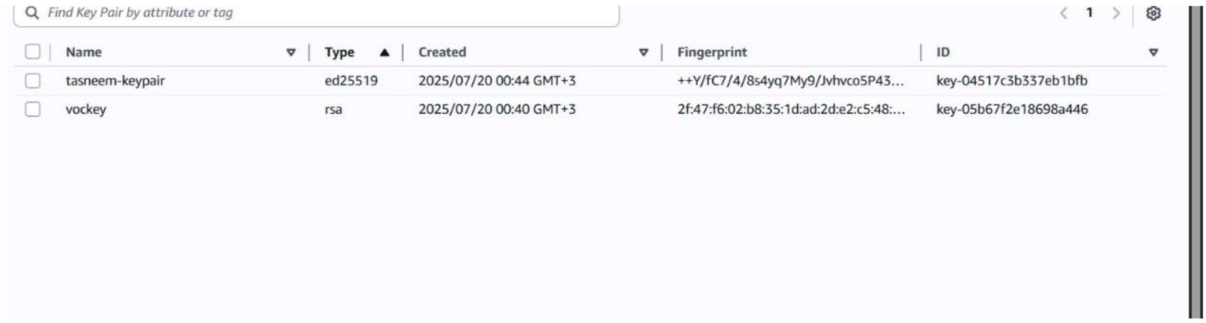
- User creation

- Package installation

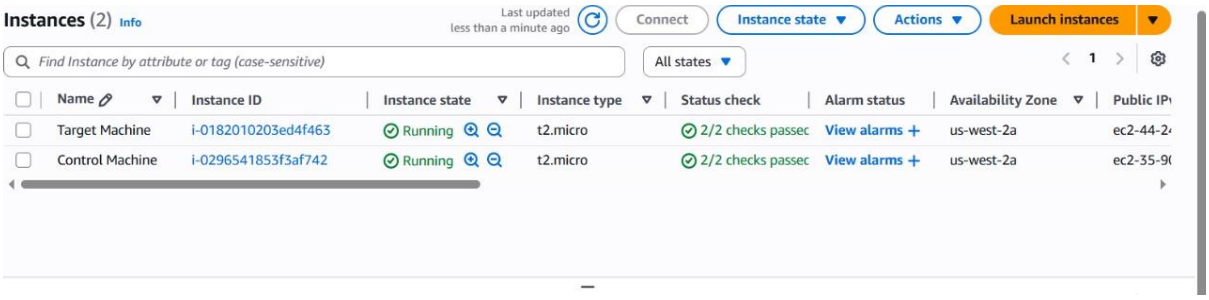
- File management

- Variables

- Facts gathering



Name	Type	Created	Fingerprint	ID
tasneem-keypair	ed25519	2025/07/20 00:44 GMT+3	++Y/fC7/4/8s4yq7My9/Jvhvco5P43...	key-04517c3b337eb1bfb
vockey	rsa	2025/07/20 00:40 GMT+3	2f:47:f6:02:b8:35:1d:ad:2d:e2:c5:48:...	key-05b67f2e18698a446



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP\
Target Machine	i-0182010203ed4f463	Running	t2.micro	2/2 checks passec	View alarms +	us-west-2a	ec2-44-2
Control Machine	i-0296541853f3af742	Running	t2.micro	2/2 checks passec	View alarms +	us-west-2a	ec2-35-9(

scp -i tasneem-keypair.pem tasneem-keypair.pem ubuntu@35.90.207.147

ssh -i tasneem-keypair.pem ubuntu@44.244.59.107

ssh -i tasneem-keypair.pem ubuntu@35.90.207.147

## ✅ Task Requirements

### 1. 🔐 Configure SSH Between Two Machines

- Set up **one control machine** (with Ansible) and **one target machine**.

- On the **control machine**, create a user called: `ansible_control`

- On the **target machine**, create a user called: `deploy`

- Ensure the control machine can SSH into the target machine using the user `deploy` **without typing a password** or typing **'yes' for confirmation**.

For control machine:

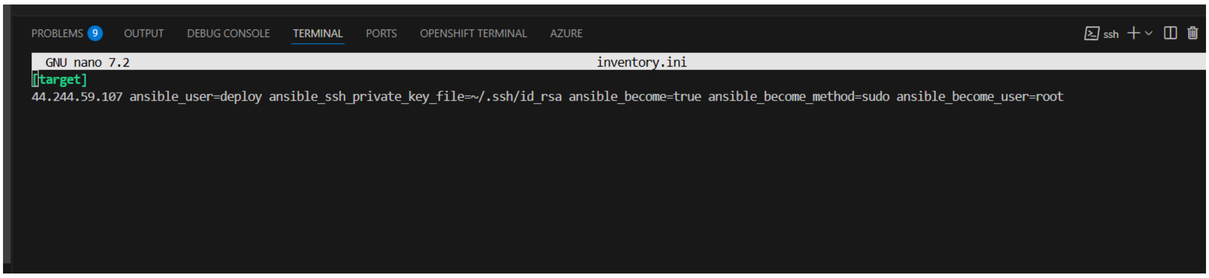
sudo adduser ansible_control

sudo usermod -aG sudo ansible_control

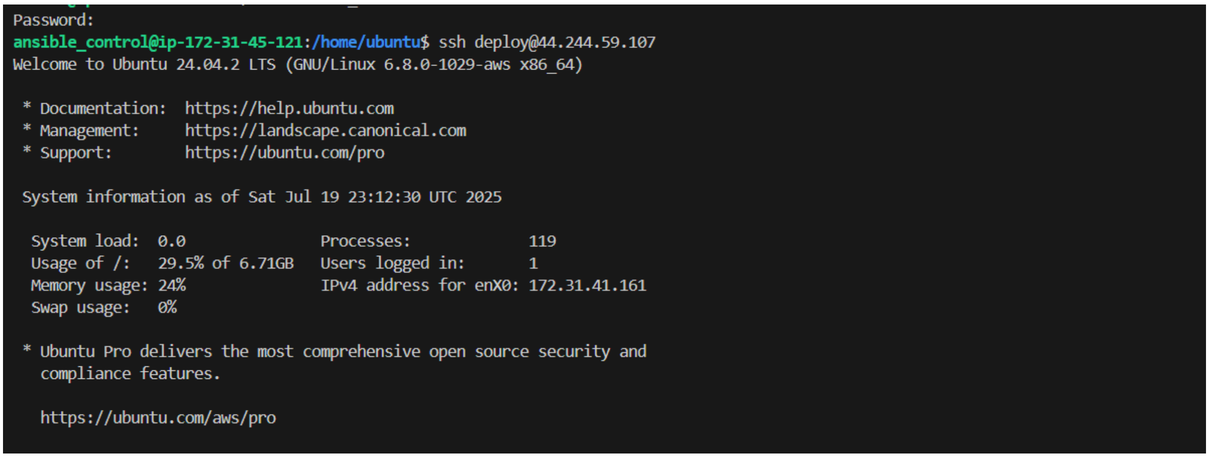
For target machine:

sudo adduser deploy

sudo usermod -aG sudo deploy



```
  GNU nano 7.2                                                    inventory.ini
[target]
44.244.59.107 ansible_user=deploy ansible_ssh_private_key_file=~/.ssh/id_rsa ansible_become=true ansible_become_method=sudo ansible_become_user=root
```



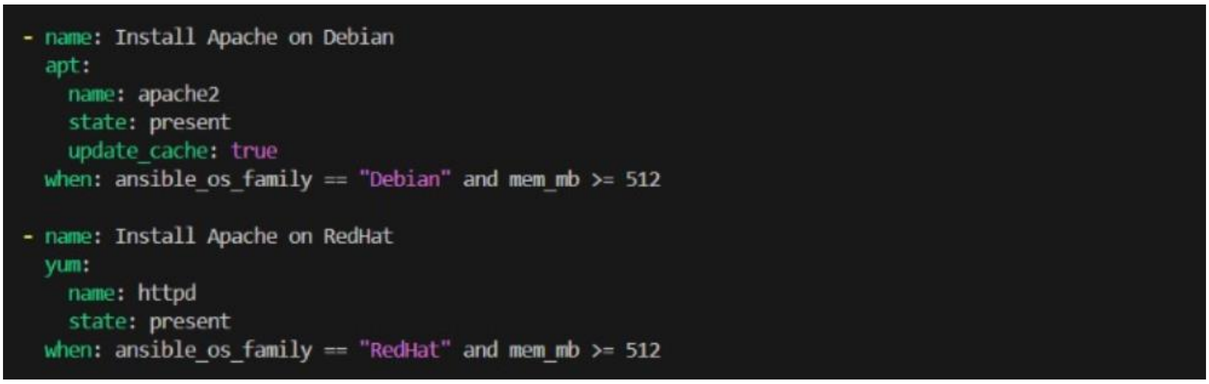
```
Password:
ansible_control@ip-172-31-45-121:/home/ubuntu$ ssh deploy@44.244.59.107
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

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

 System information as of Sat Jul 19 23:12:30 UTC 2025

  System load:  0.0               Processes:             119
  Usage of /:   29.5% of 6.71GB   Users logged in:       1
  Memory usage: 24%               IPv4 address for enX0: 172.31.41.161
  Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro
```

### 2. 📦 Install Apache (Latest Version) on the Target Machine

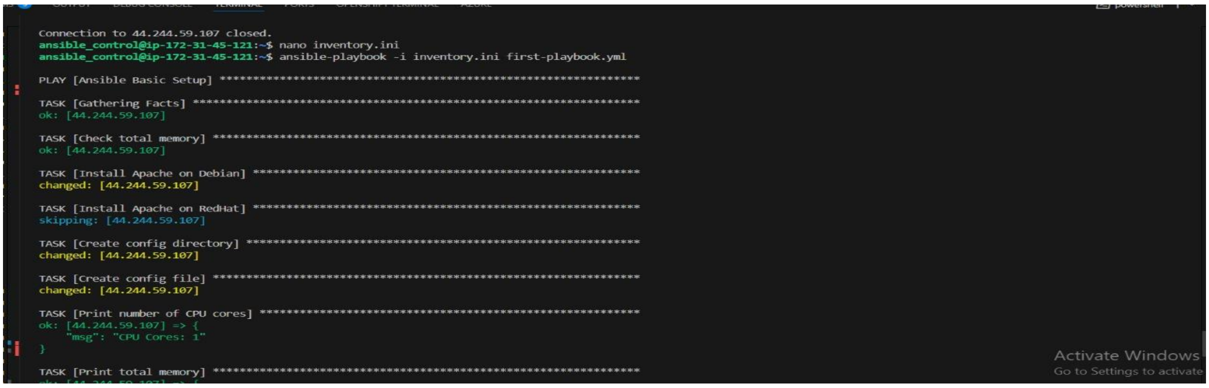
- Install `apache2` on Debian or `httpd` on RedHat.

- Use `when` condition based on `ansible_os_family`.

- Skip the install if total memory < 512MB using .



```
- name: Install Apache on Debian
  apt:
    name: apache2
    state: present
    update_cache: true
  when: ansible_os_family == "Debian" and mem_mb >= 512

- name: Install Apache on RedHat
  yum:
    name: httpd
    state: present
  when: ansible_os_family == "RedHat" and mem_mb >= 512
```



```
Connection to 44.244.59.107 closed.
ansible_control@ip-172-31-45-121:~$ nano inventory.ini
ansible_control@ip-172-31-45-121:~$ ansible-playbook -i inventory.ini first-playbook.yml

PLAY [Ansible Basic Setup] ***************************************************************

TASK [Gathering Facts] *******************************************************************
ok: [44.244.59.107]

TASK [Check total memory] ****************************************************************
ok: [44.244.59.107]

TASK [Install Apache on Debian] **********************************************************
changed: [44.244.59.107]

TASK [Install Apache on RedHat] **********************************************************
skipping: [44.244.59.107]

TASK [Create config directory] ***********************************************************
changed: [44.244.59.107]

TASK [Create config file] ****************************************************************
changed: [44.244.59.107]

TASK [Print number of CPU cores] *********************************************************
ok: [44.244.59.107] => {
    "msg": "CPU Cores: 1"
}

TASK [Print total memory] ****************************************************************
```

### 3. 📁 Create a Directory and Config File

- Create a directory: `/etc/demo_config`

- Inside it, create a file: `config`

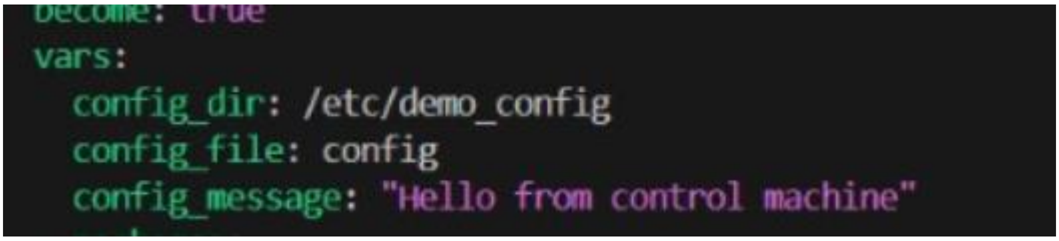
- Add the line: `"Hello from control machine"`

🔧 Use variables for:

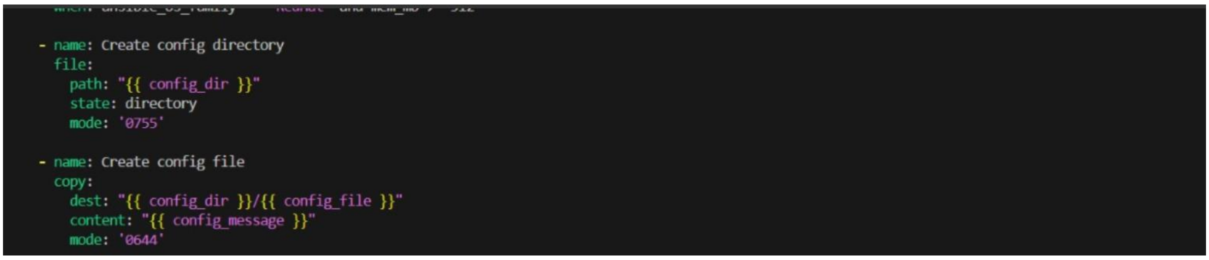
- The directory path

- The file name

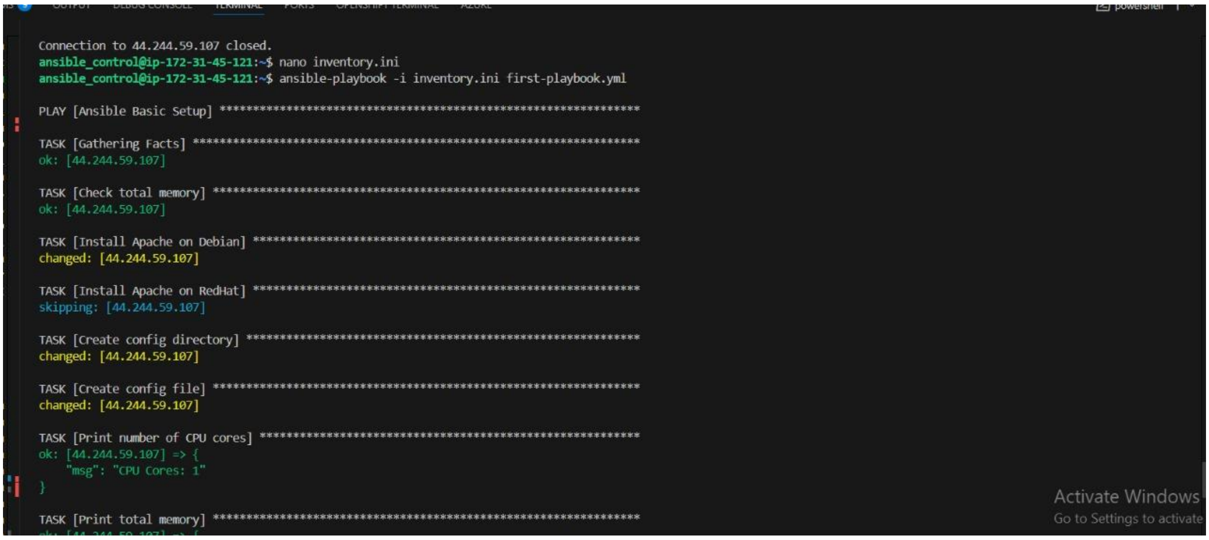
- The message



```
become: true
vars:
  config_dir: /etc/demo_config
  config_file: config
  config_message: "Hello from control machine"
```



```
- name: Create config directory
  file:
    path: "{{ config_dir }}"
    state: directory
    mode: '0755'

- name: Create config file
  copy:
    dest: "{{ config_dir }}/{{ config_file }}"
    content: "{{ config_message }}"
    mode: '0644'
```

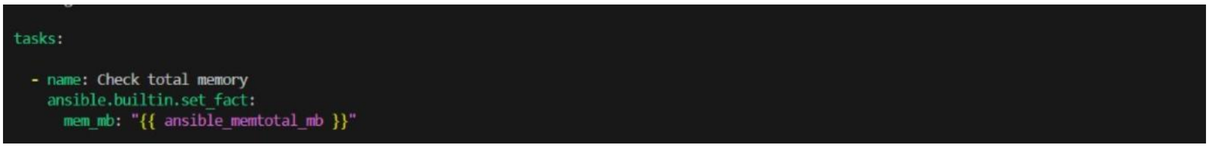


```
Connection to 44.244.59.107 closed.
ansible_control@ip-172-31-45-121:~$ nano inventory.ini
ansible_control@ip-172-31-45-121:~$ ansible-playbook -i inventory.ini first-playbook.yml

PLAY [Ansible Basic Setup] ****************************************************************

TASK [Gathering Facts] ********************************************************************
ok: [44.244.59.107]

TASK [Check total memory] *****************************************************************
ok: [44.244.59.107]

TASK [Install Apache on Debian] ***********************************************************
changed: [44.244.59.107]

TASK [Install Apache on RedHat] ***********************************************************
skipping: [44.244.59.107]

TASK [Create config directory] ************************************************************
changed: [44.244.59.107]

TASK [Create config file] *****************************************************************
changed: [44.244.59.107]

TASK [Print number of CPU cores] **********************************************************
ok: [44.244.59.107] => {
    "msg": "CPU Cores: 1"
}

TASK [Print total memory] *****************************************************************
ok: [44.244.59.107] => {
```

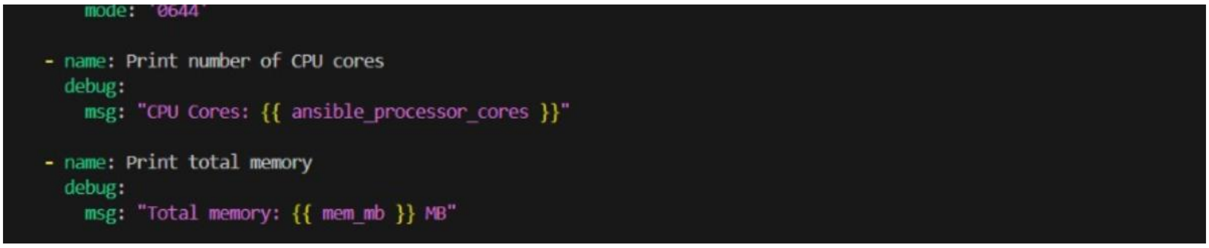
### 4. 🧠 Print System Facts

Use the `debug` module to print:

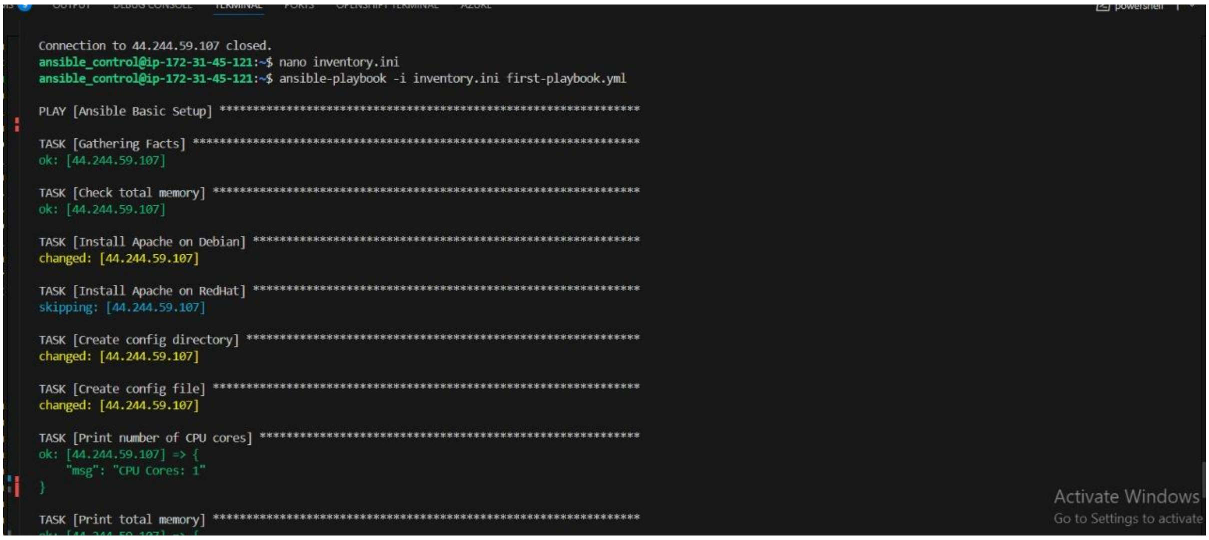
- 🧮 Number of CPU cores
- 🧠 Total memory in MB



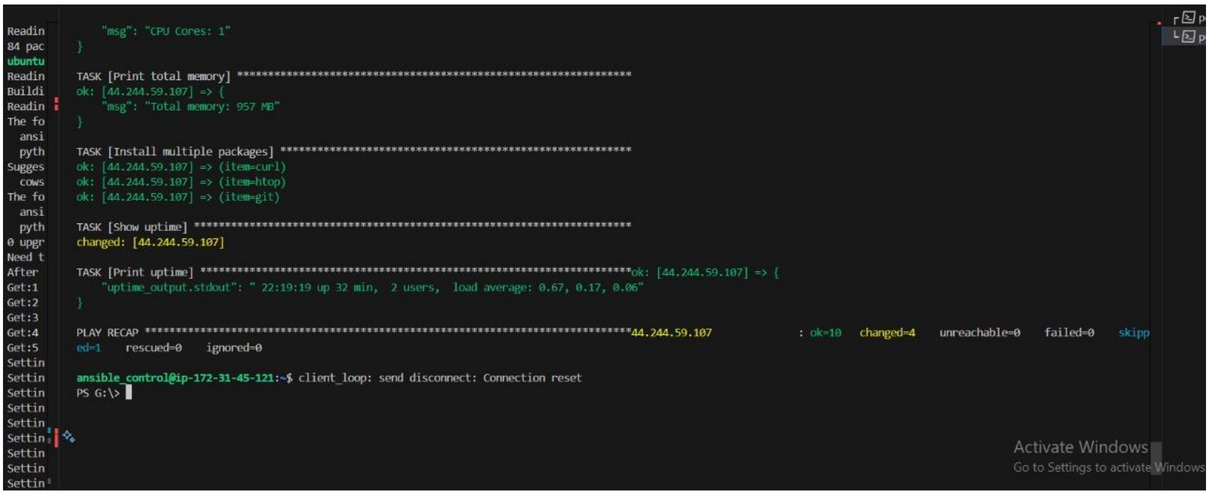
```
tasks:

  - name: Check total memory
    ansible.builtin.set_fact:
      mem_mb: "{{ ansible_memtotal_mb }}"
```



```
        mode: '0644'

    - name: Print number of CPU cores
      debug:
        msg: "CPU Cores: {{ ansible_processor_cores }}"

    - name: Print total memory
      debug:
        msg: "Total memory: {{ mem_mb }} MB"
```



```
Connection to 44.244.59.107 closed.
ansible_control@ip-172-31-45-121:~$ nano inventory.ini
ansible_control@ip-172-31-45-121:~$ ansible-playbook -i inventory.ini first-playbook.yml

PLAY [Ansible Basic Setup] ***************************************************************

TASK [Gathering Facts] *******************************************************************
ok: [44.244.59.107]

TASK [Check total memory] ****************************************************************
ok: [44.244.59.107]

TASK [Install Apache on Debian] **********************************************************
changed: [44.244.59.107]

TASK [Install Apache on RedHat] **********************************************************
skipping: [44.244.59.107]

TASK [Create config directory] ***********************************************************
changed: [44.244.59.107]

TASK [Create config file] ****************************************************************
changed: [44.244.59.107]

TASK [Print number of CPU cores] *********************************************************
ok: [44.244.59.107] => {
    "msg": "CPU Cores: 1"
}

TASK [Print total memory] ****************************************************************
```



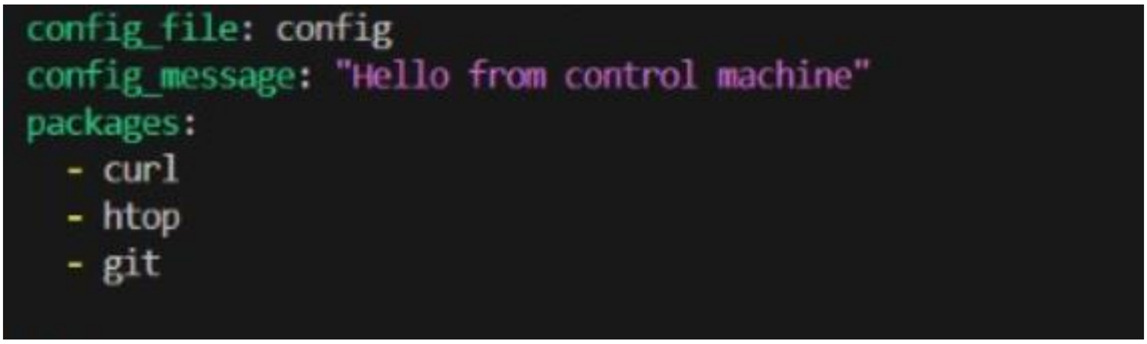
```
Readin
84 pac
ubuntu
Readin
Buildi
Readin
The fo
  ansi
  pyth
Sugges
  cows
The fo
  ansi
  pyth
0 upgr
Need t
After
Get:1
Get:2
Get:3
Get:4
Get:5
Settin
Settin
Settin
Settin
Settin
Settin
Settin
Settin
Settin
```

```
    "msg": "CPU Cores: 1"
}

TASK [Print total memory] ****************************************************************
ok: [44.244.59.107] => {
    "msg": "Total memory: 957 MB"
}

TASK [Install multiple packages] *********************************************************
ok: [44.244.59.107] => (item=curl)
ok: [44.244.59.107] => (item=htop)
ok: [44.244.59.107] => (item=git)

TASK [Show uptime] ***********************************************************************
changed: [44.244.59.107]

TASK [Print uptime] **********************************************************************ok: [44.244.59.107] => {
    "uptime_output.stdout": " 22:19:19 up 32 min,  2 users,  load average: 0.67, 0.17, 0.06"
}

PLAY RECAP *******************************************************************************44.244.59.107              : ok=10   changed=4    unreachable=0    failed=0    skipp
ed=1    rescued=0    ignored=0

ansible_control@ip-172-31-45-121:~$ client_loop: send disconnect: Connection reset
PS G:\>
```

### 5. 🔁 Install Multiple Packages Using Loop

Define Var:

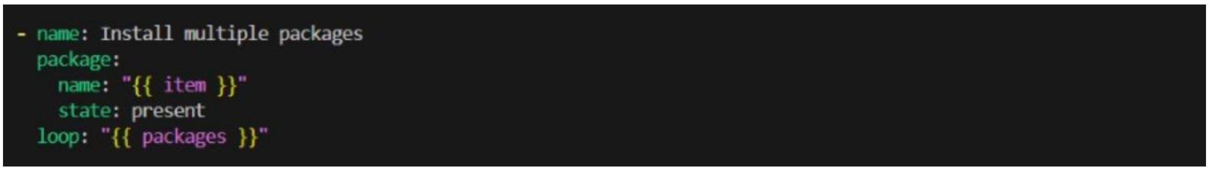
```yaml

packages:

  - curl

  - htop

  - git
```



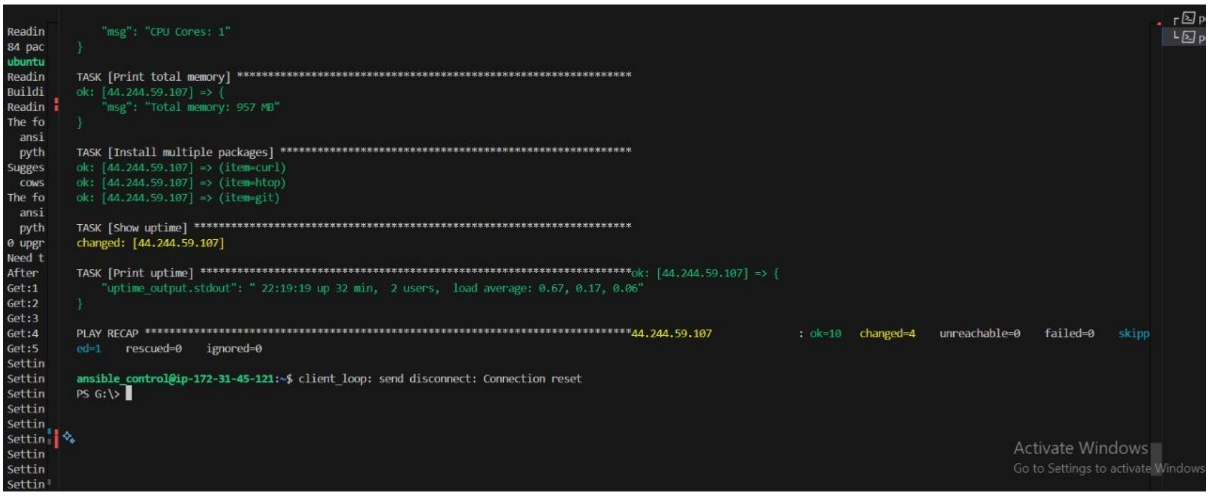
```yaml
config_file: config
config_message: "Hello from control machine"
packages:
  - curl
  - htop
  - git
```

` ` `

Use ` loop` to install each of them.



```yaml
- name: Install multiple packages
  package:
    name: "{{ item }}"
    state: present
  loop: "{{ packages }}"
```



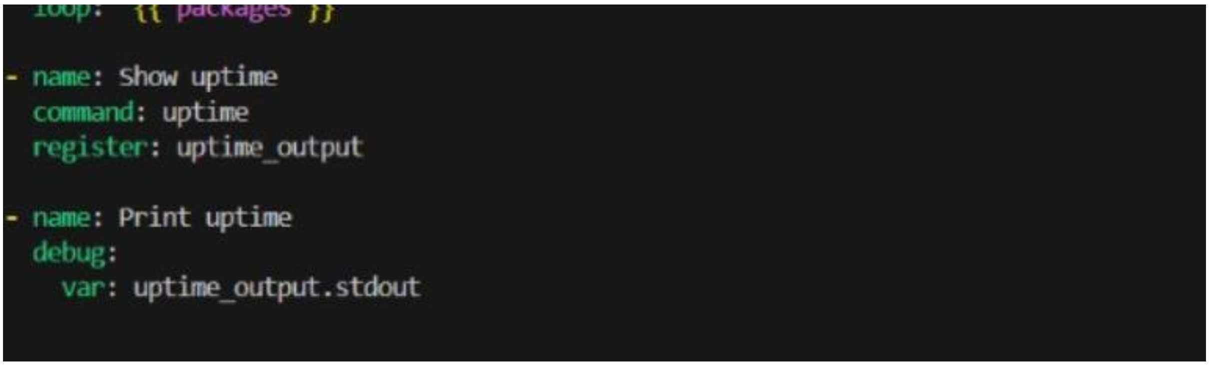
```
    "msg": "CPU Cores: 1"
}

TASK [Print total memory] *********************************************************************
ok: [44.244.59.107] => {
    "msg": "Total memory: 957 MB"
}

TASK [Install multiple packages] **************************************************************
ok: [44.244.59.107] => (item=curl)
ok: [44.244.59.107] => (item=htop)
ok: [44.244.59.107] => (item=git)

TASK [Show uptime] ****************************************************************************
changed: [44.244.59.107]

TASK [Print uptime] ***************************************************************************ok: [44.244.59.107] => {
    "uptime_output.stdout": " 22:19:19 up 32 min,  2 users,  load average: 0.67, 0.17, 0.06"
}

PLAY RECAP ************************************************************************************44.244.59.107              : ok=10   changed=4    unreachable=0    failed=0    skipp
ed=1    rescued=0    ignored=0

ansible_control@ip-172-31-45-121:~$ client_loop: send disconnect: Connection reset
PS G:\>
```

```
Readin
84 pac
ubuntu
Readin
Buildi
Readin
The fo
  ansi
  pyth
Sugges
  cows
The fo
  ansi
  pyth
0 upgr
Need t
After
Get:1
Get:2
Get:3
Get:4
Get:5
Settin
Settin
Settin
Settin
Settin
Settin
Settin
Settin
Settin
```

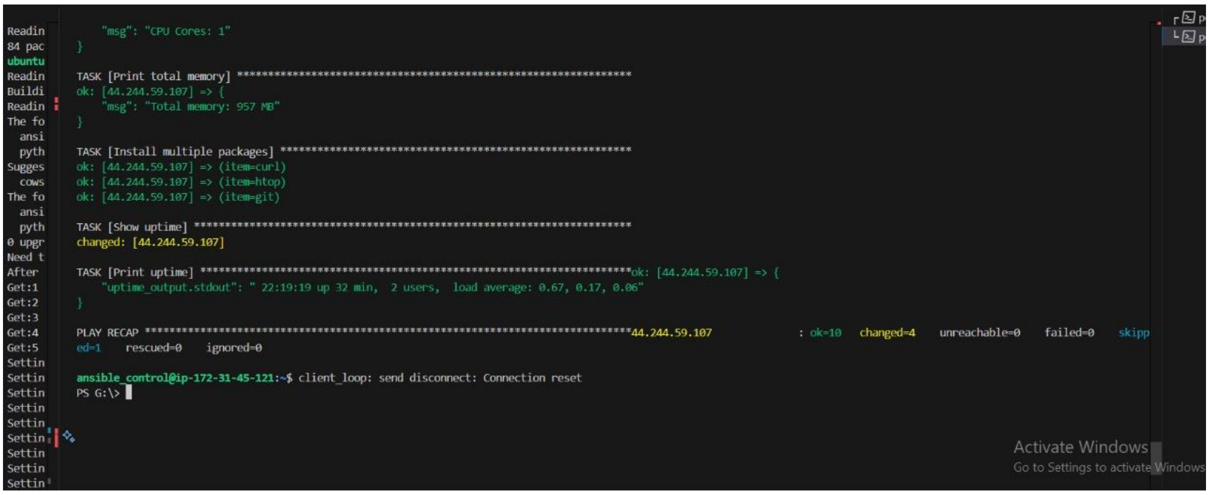
### 6. ⏱️ Show Uptime

- Use the ` command` module to run ` uptime` .

- Print the output in terminal



```
  loop: "{{ packages }}"

- name: Show uptime
  command: uptime
  register: uptime_output

- name: Print uptime
  debug:
    var: uptime_output.stdout
```



```
Readin       "msg": "CPU Cores: 1"
84 pac   }
ubuntu
Readin   TASK [Print total memory] *****************************************************************
Buildi   ok: [44.244.59.107] => {
Readin       "msg": "Total memory: 957 MB"
The fo   }
  ansi
  pyth   TASK [Install multiple packages] **********************************************************
Sugges   ok: [44.244.59.107] => (item=curl)
  cows   ok: [44.244.59.107] => (item=htop)
The fo   ok: [44.244.59.107] => (item=git)
  ansi
  pyth   TASK [Show uptime] ************************************************************************
0 upgr   changed: [44.244.59.107]
Need t
After    TASK [Print uptime] ***********************************************************************ok: [44.244.59.107] => {
Get:1        "uptime_output.stdout": " 22:19:19 up 32 min,  2 users,  load average: 0.67, 0.17, 0.06"
Get:2    }
Get:3
Get:4    PLAY RECAP ********************************************************************************44.244.59.107              : ok=10    changed=4    unreachable=0    failed=0    skipp
Get:5    ed=1    rescued=0    ignored=0
Settin
Settin   ansible_control@ip-172-31-45-121:~$ client_loop: send disconnect: Connection reset
Settin   PS G:\>
Settin
Settin
Settin
Settin
Settin
Settin
```
