**Ansible-02**

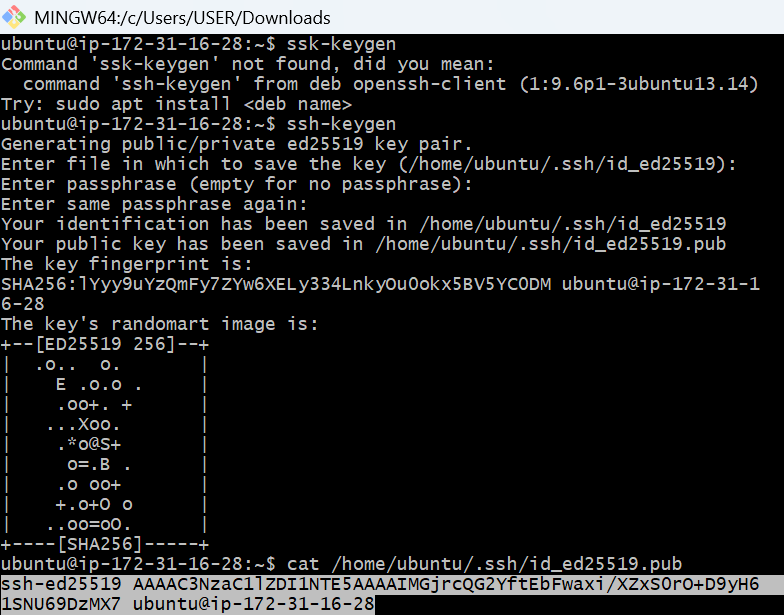
**1.Watch ansible-02 video and write down notes.**

**2.Install httpd using ansible playbook, use handlers, notifiers.**

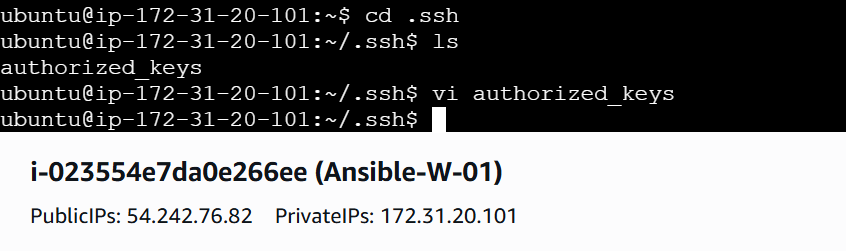
**Step 1: Generate and Share SSH Keys**

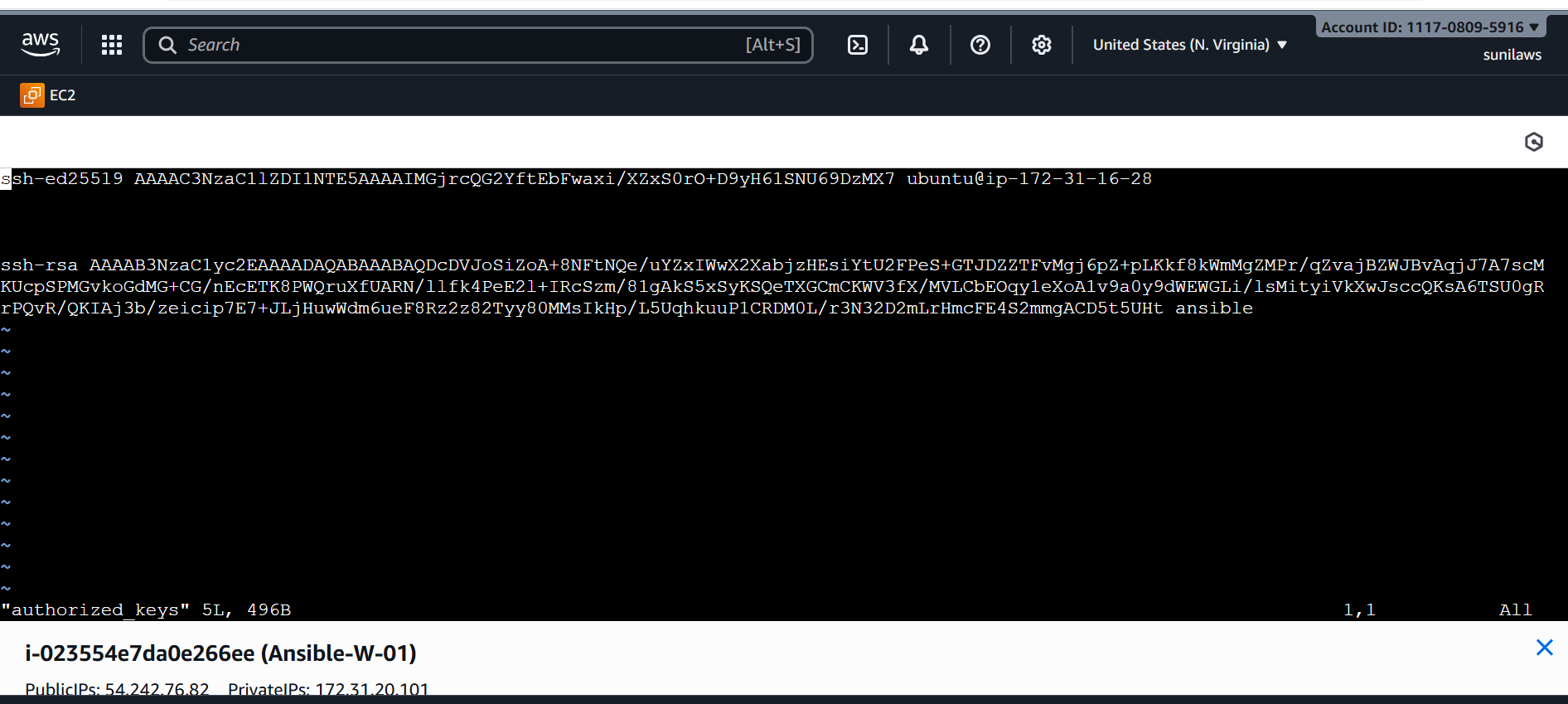
On your **control node** (Ubuntu instance where Ansible is installed):

Do ssh-keygen and copy the key



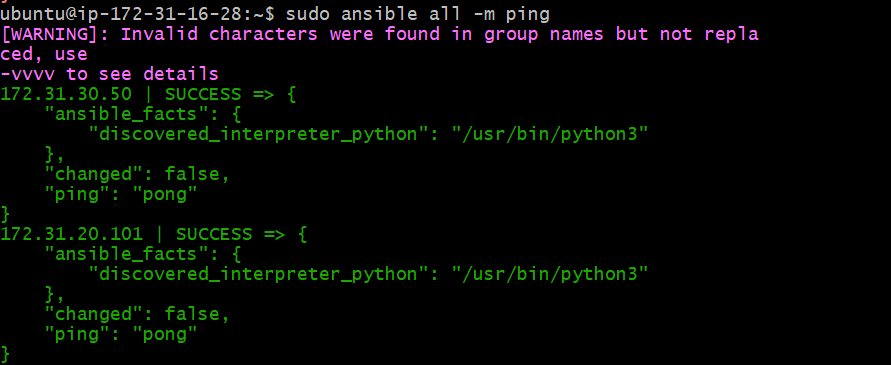
Now copy the public key to your target nodes: like worker 1 and worker2





Check the ping from controller ec2

sudo ansible all -m ping



**Step 2: Create Playbook Folder**

mkdir ~/playbooks

cd ~/playbooks

Inside, create your playbook file:

Vi apache.yml

**Step 3: Write the Playbook (apache.yml)**

---

- hosts: all

become: yes

remote\_user: ubuntu

tasks:

- name: Install Apache

apt:

name: apache2

state: latest

update\_cache: yes

- name: Copy index.html

copy:

src: index.html

dest: /var/www/html/index.html

notify: Restart apache2 # <-- aligned with copy, not inside it

handlers:

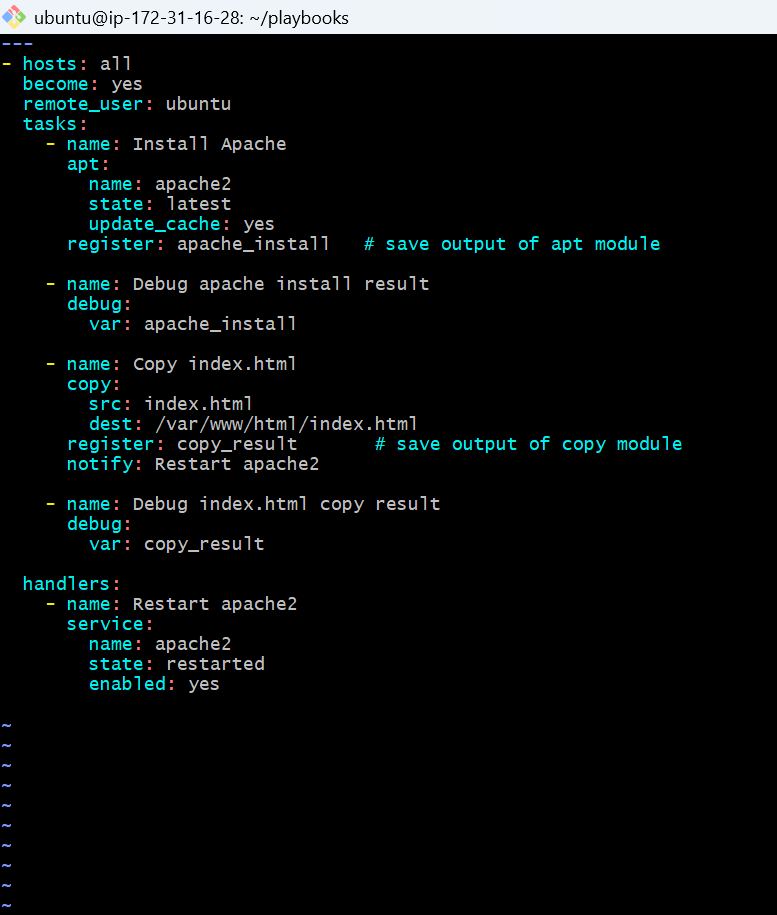
- name: Restart apache2

service:

name: apache2

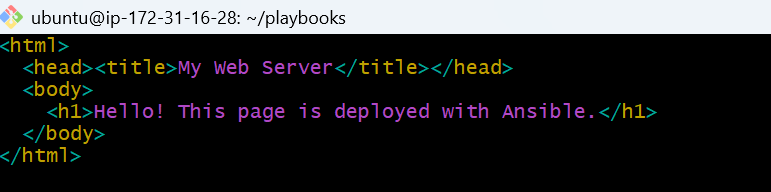
state: restarted

enabled: yes



**Step 4: Create index.html File**

Vi index.html



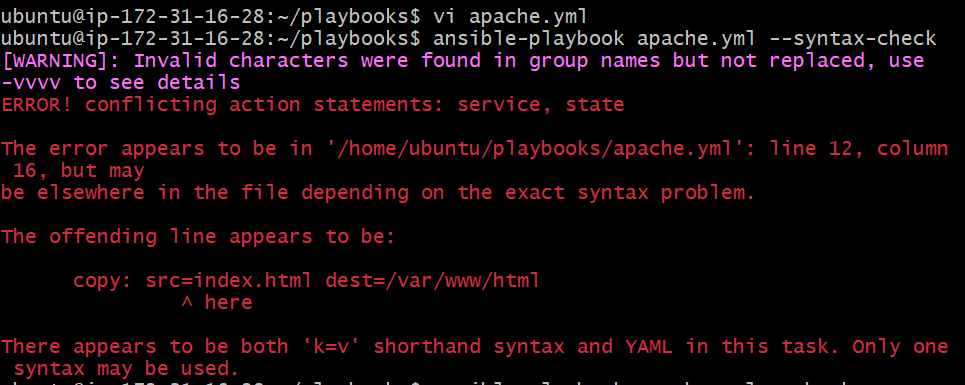
**Step 5: Validate Playbook Before Running**

From inside ~/playbooks:

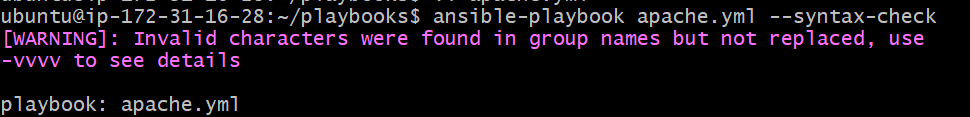
✅ Check YAML syntax

ansible-playbook apache.yml --syntax-check

If any syntax errors it appers like this then modify the code and run

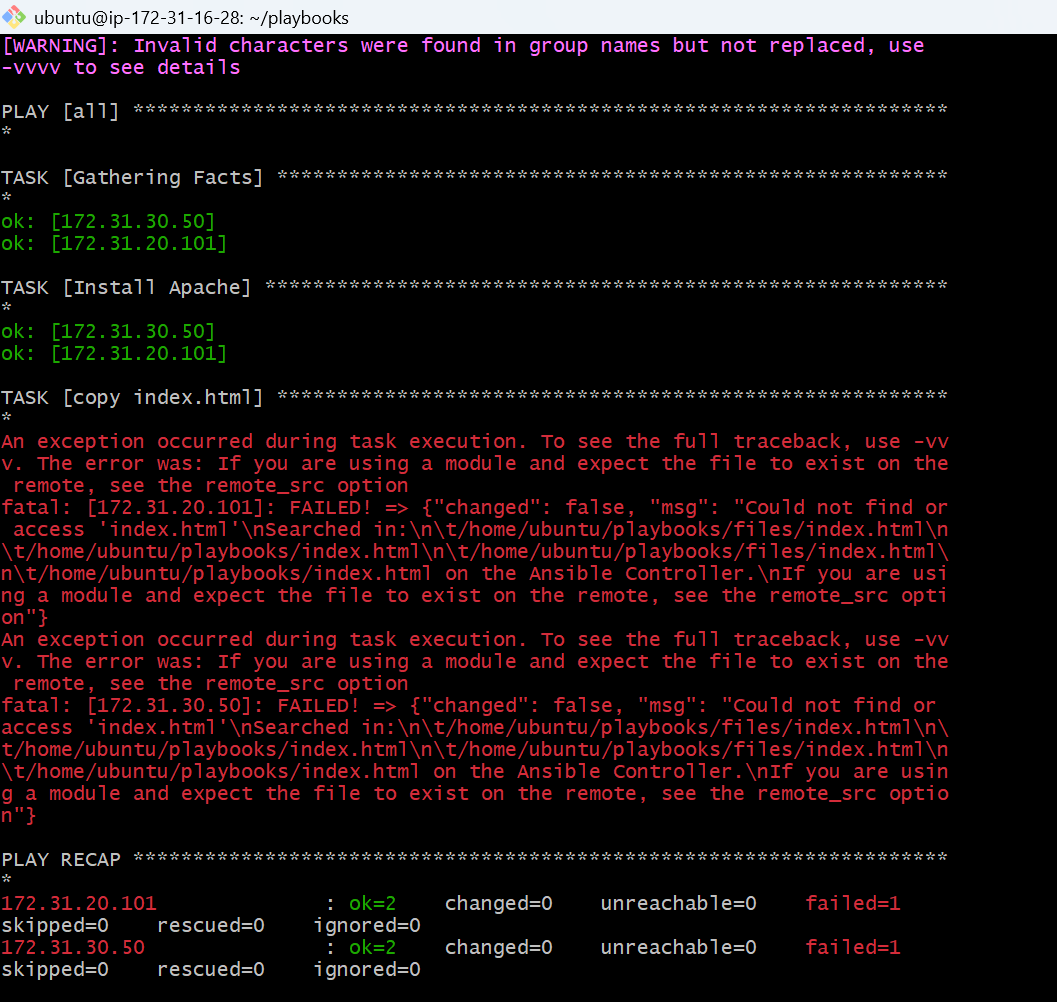


If no syntax errors



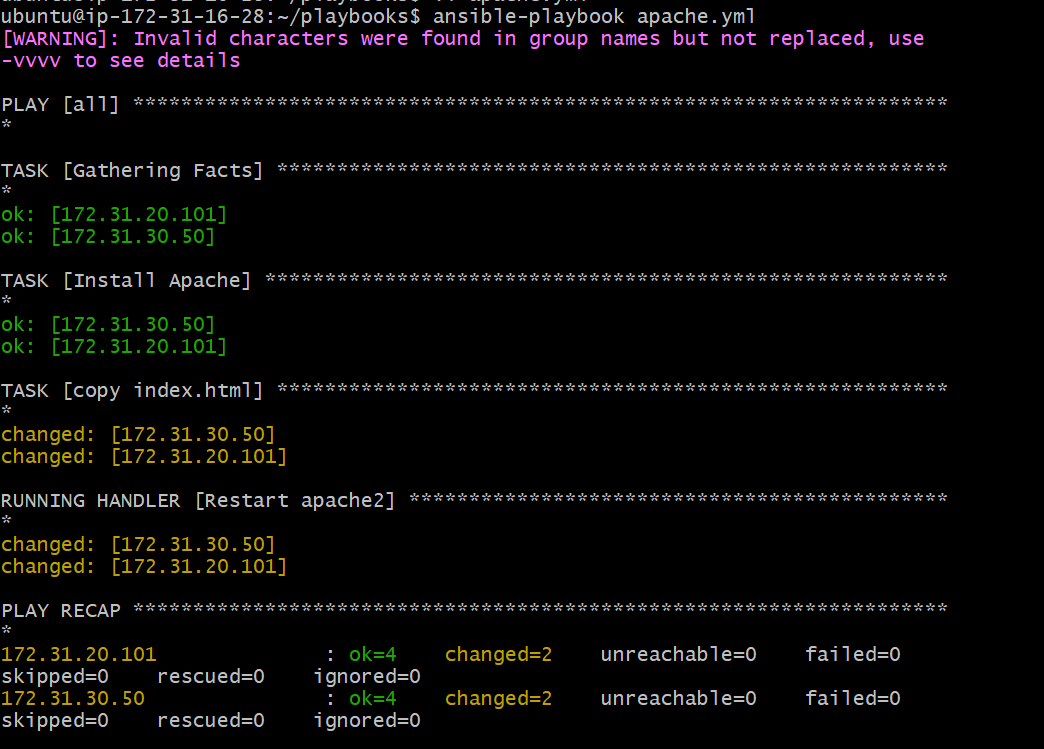
Dry run (no changes, just simulation)

ansible-playbook apache.yml --check

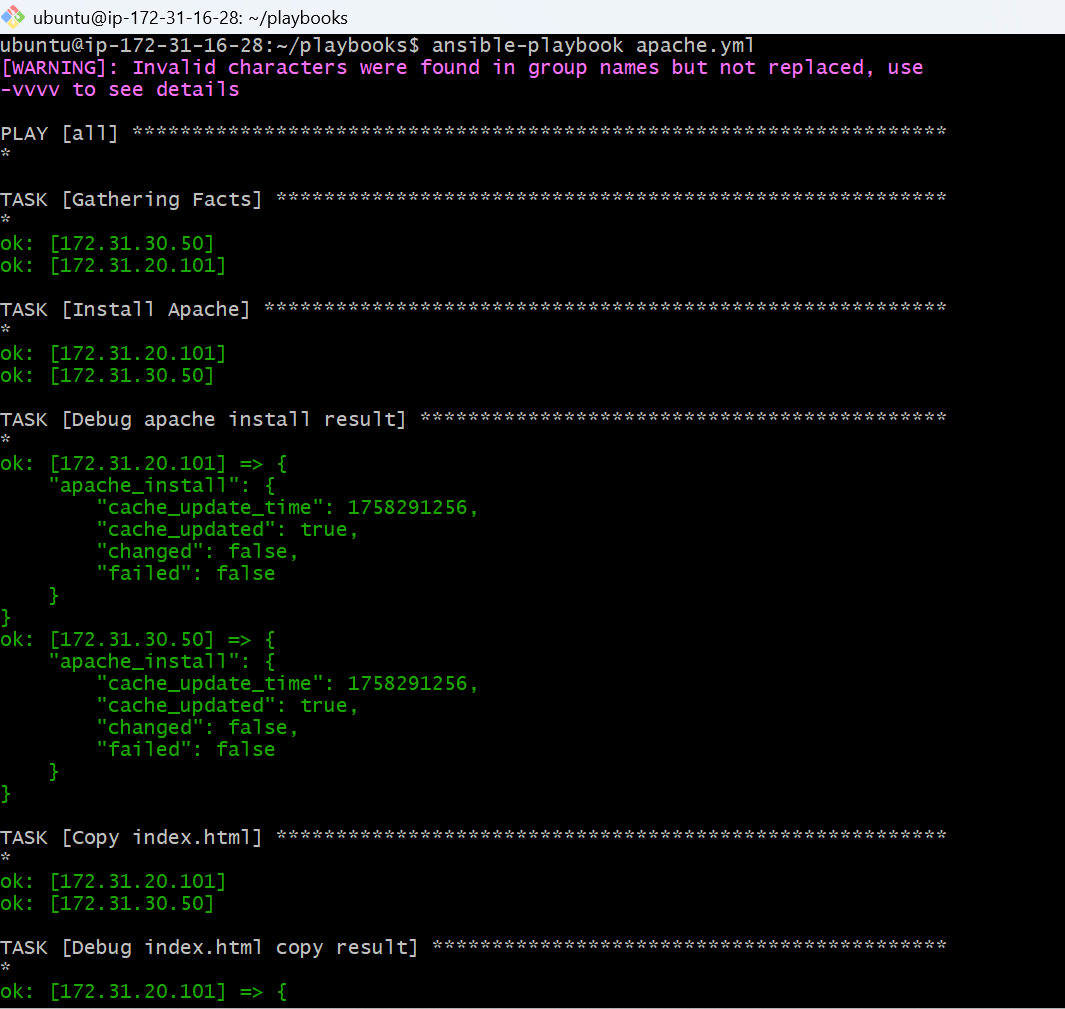


**Step -6 : Run playbook for real**

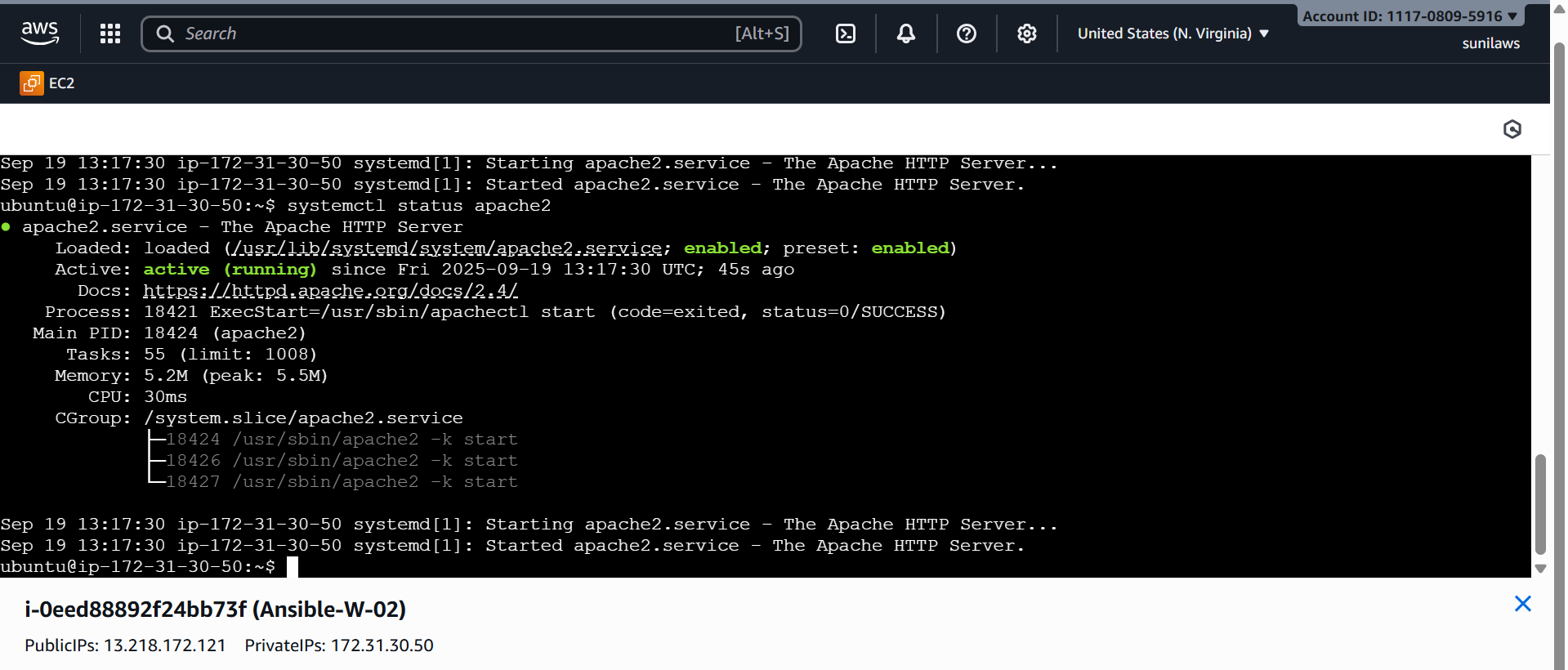
ansible-playbook apache.yml



(optional)We can add debug and register module also

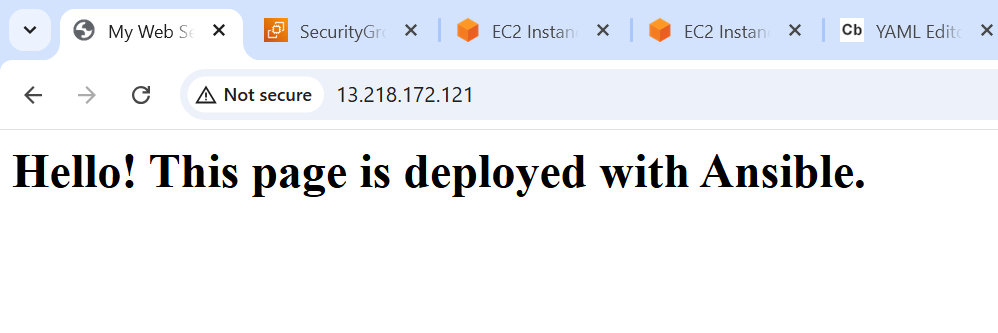


**Step 7: Verify Installation**

****

Check in browser:

http://<your-target-ip**> worker public ip**



### ****Modules****

* **apt**→ Manages packages on Ubuntu/Amazon Linux.
* apt :
* name: httpd
* state: present

Installs httpd if not present.

* **copy** → Copies files from control node → target node.
* copy:
* src: index.html
* dest: /var/www/html/index.html
* **service** → Manages services (start, stop, restart).
* service:
* name: httpd
* state: restarted

### ****Handlers & Notify****

* **notify** → Tells Ansible to trigger a handler if the task changes something.
* **Handlers** → Special tasks that run at the end of the playbook, only if notified.

Example:

* If httpd is installed → notify: Start httpd.
* If index.html changes → notify: Restart httpd.

**3.Write a ansible playbook to install Apache tomcat.**

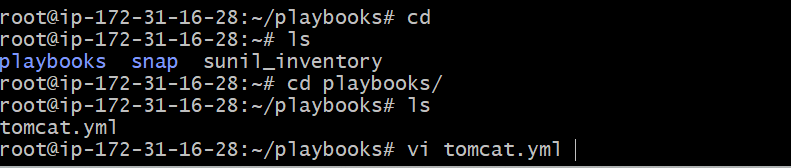
**Goto root**

**Create Ansible project folder**

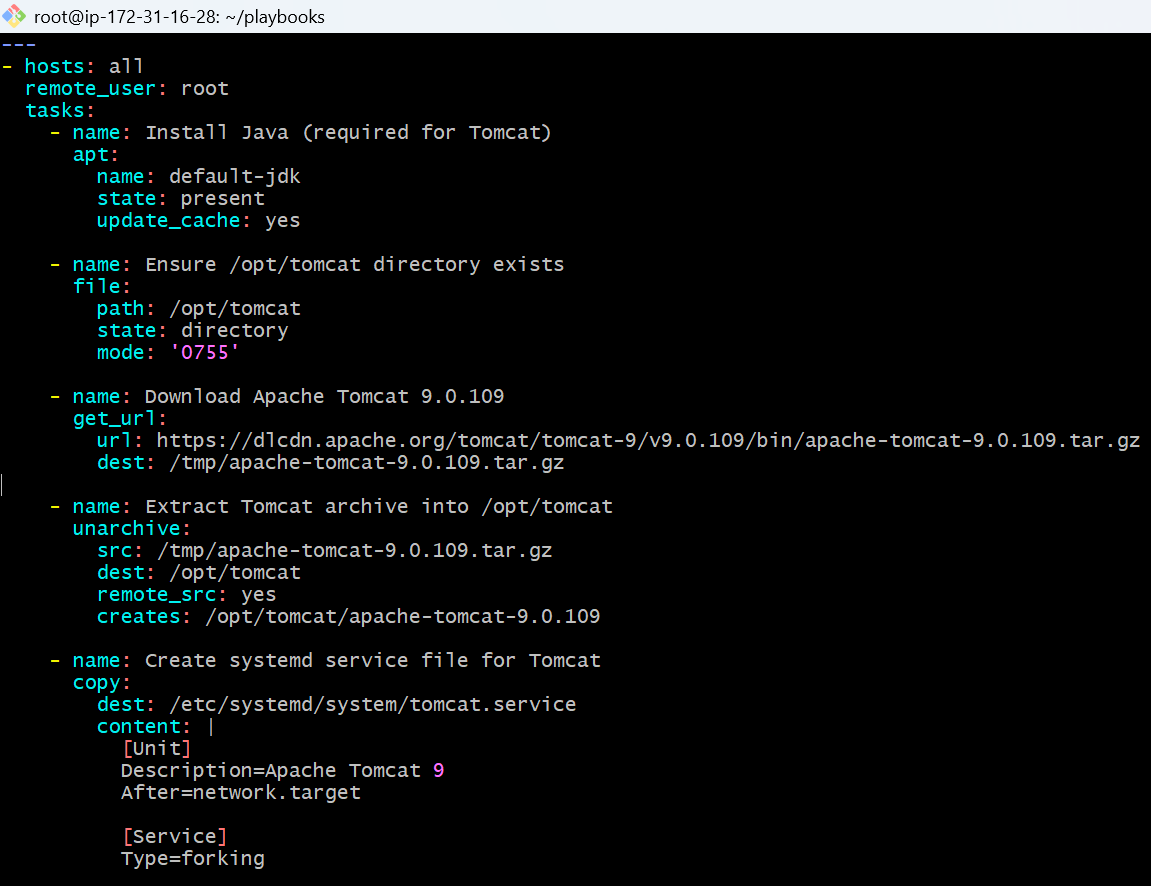
**mkdir ~/playbooks**

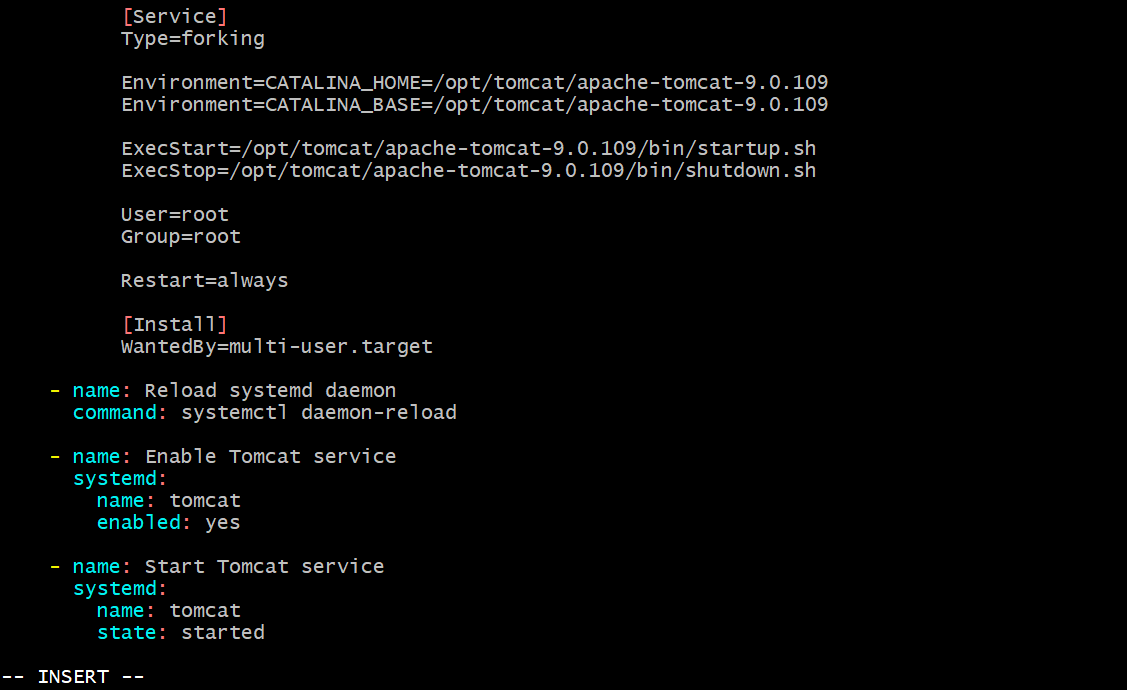
**cd ~/playbooks**

**. Write Playbook (tomcat.yml)**

****

**Vi tomcat.yml**

****

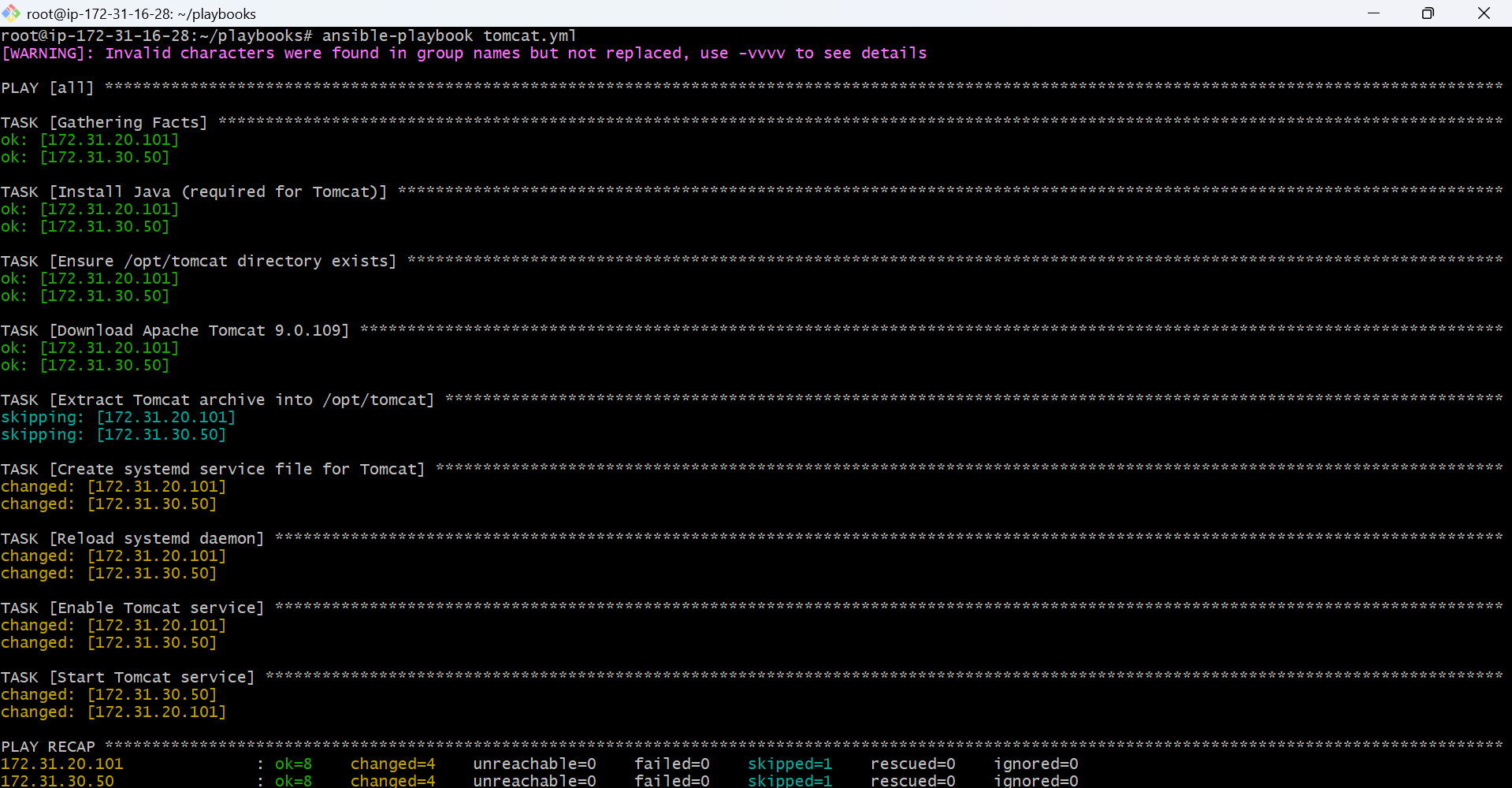
****

**Check before run**

**ansible-playbook tomcat.yml --syntax-check**

**ansible-playbook tomcat.yml --check**

**ansible-playbook tomcat.yml**

****

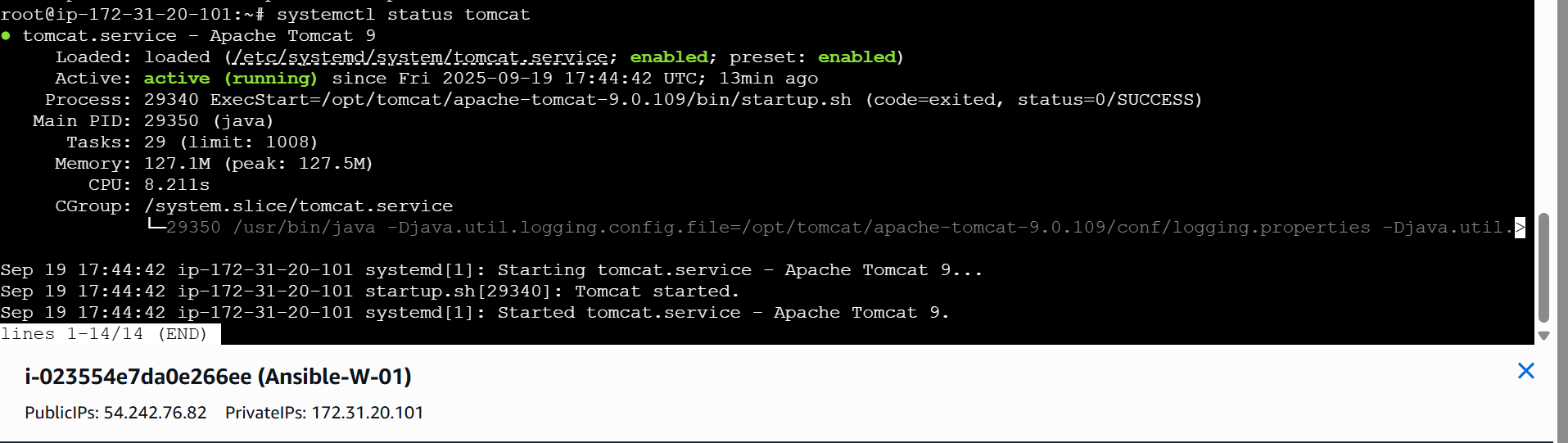
## Key Modules Used

* **apt → Installs Java**
* **file → Creates /opt/tomcat directory**
* **get\_url → Downloads Tomcat archive**
* **unarchive → Extracts Tomcat into /opt/tomcat**
* **copy → Creates tomcat.service systemd file**
* **command → Reloads systemd daemon**
* **systemd → Enables and starts Tomcat**

### ****Verify Tomcat service****

**On target server:**

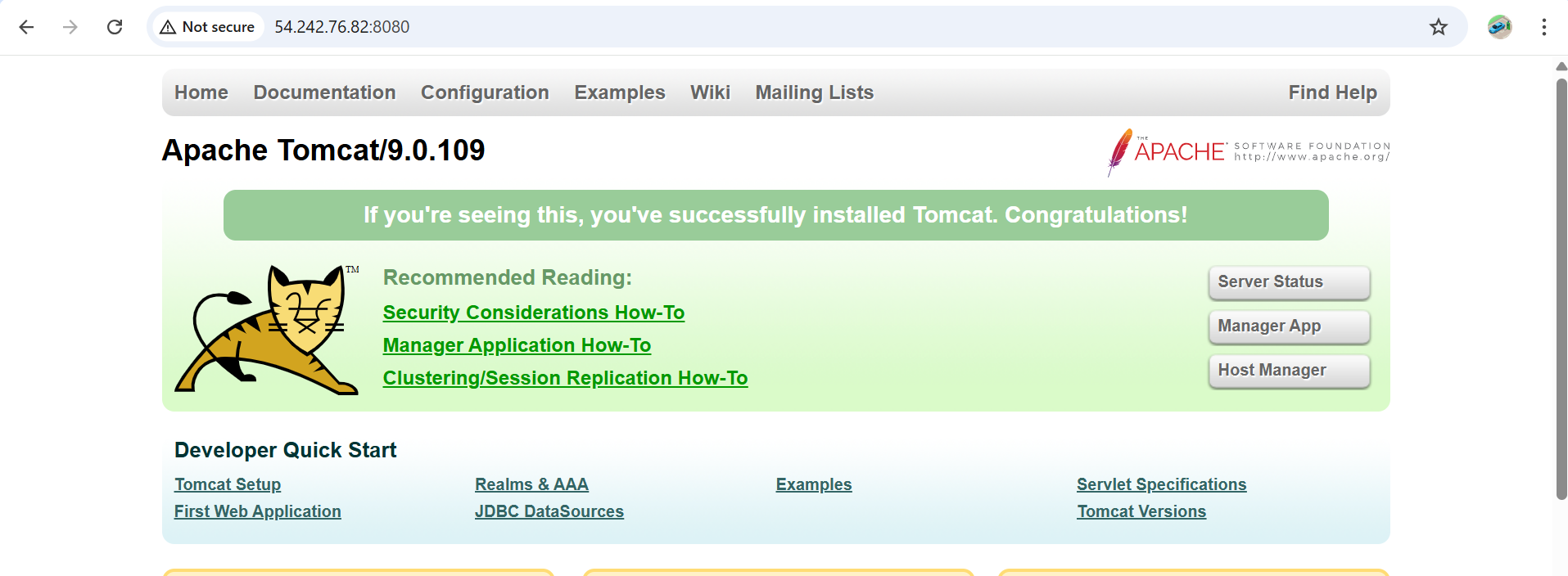
**systemctl status tomcat**

****

### ****Test in browser****

**Go to:**

**http://<server-ip>:8080**

****

**---**

**- hosts: all**

**remote\_user: root**

**tasks:**

**- name: Install Java (required for Tomcat)**

**apt:**

**name: default-jdk**

**state: present**

**update\_cache: yes**

**- name: Ensure /opt/tomcat directory exists**

**file:**

**path: /opt/tomcat**

**state: directory**

**mode: '0755'**

**- name: Download Apache Tomcat 9.0.109**

**get\_url:**

**url: https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.109/bin/apache-tomcat-9.0.109.tar.gz**

**dest: /tmp/apache-tomcat-9.0.109.tar.gz**

**- name: Extract Tomcat archive into /opt/tomcat**

**unarchive:**

**src: /tmp/apache-tomcat-9.0.109.tar.gz**

**dest: /opt/tomcat**

**remote\_src: yes**

**creates: /opt/tomcat/apache-tomcat-9.0.109**

**- name: Create systemd service file for Tomcat**

**copy:**

**dest: /etc/systemd/system/tomcat.service**

**content: |**

**[Unit]**

**Description=Apache Tomcat 9**

**After=network.target**

**[Service]**

**Type=forking**

**Environment=CATALINA\_HOME=/opt/tomcat/apache-tomcat-9.0.109**

**Environment=CATALINA\_BASE=/opt/tomcat/apache-tomcat-9.0.109**

**ExecStart=/opt/tomcat/apache-tomcat-9.0.109/bin/startup.sh**

**ExecStop=/opt/tomcat/apache-tomcat-9.0.109/bin/shutdown.sh**

**User=root**

**Group=root**

**Restart=always**

**[Install]**

**WantedBy=multi-user.target**

**- name: Reload systemd daemon**

**command: systemctl daemon-reload**

**- name: Enable Tomcat service**

**systemd:**

**name: tomcat**

**enabled: yes**

**- name: Start Tomcat service**

**systemd:**

**name: tomcat**

**state: started**

**4.Write a ansible playbook to provision one ec2 on aws.**

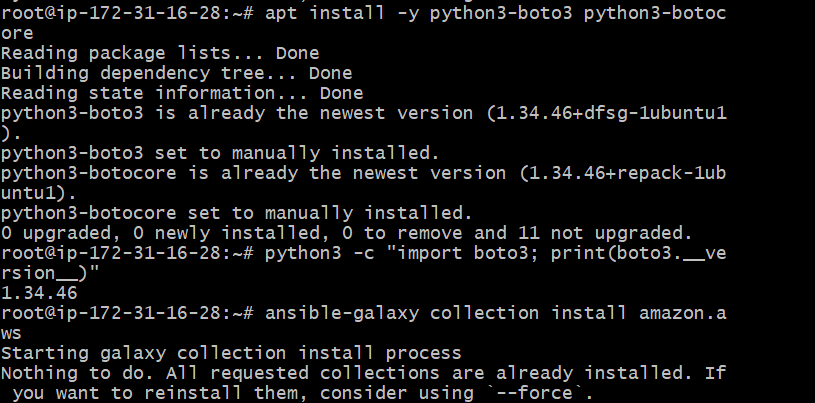
## Prerequisites

**1.Install boto3 & botocore** (Ansible needs them to talk to AWS):

apt install -y python3-boto3 python3-botocore

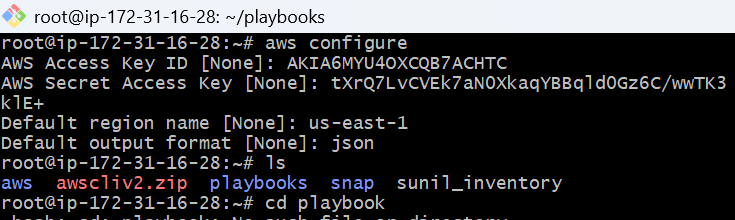
**2.Ensure Ansible collections are installed**:

ansible-galaxy collection install amazon.aws

****

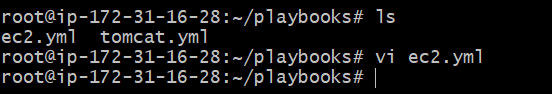
3.Configure AWS credentials

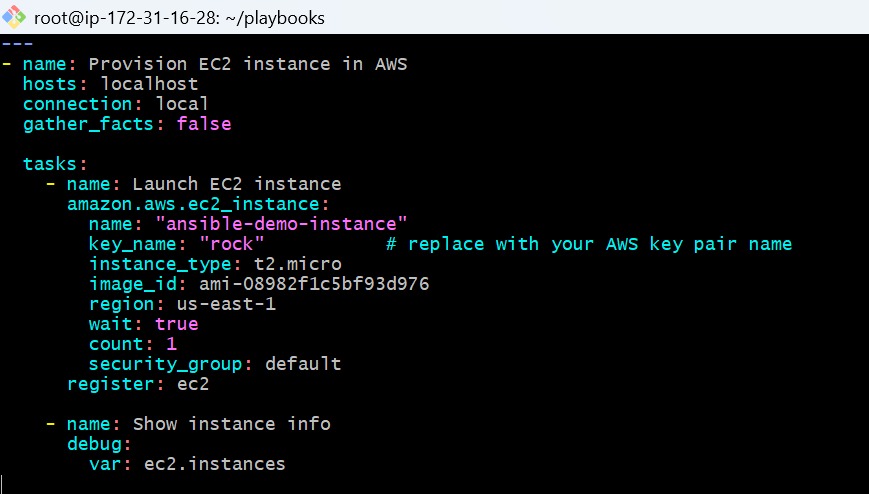
Aws configure

****

Step 4: Write your playbook (ec2.yml)

Inside your play book create ec2.yml



****

## Step 5 : Run playbook

Check syntax first:

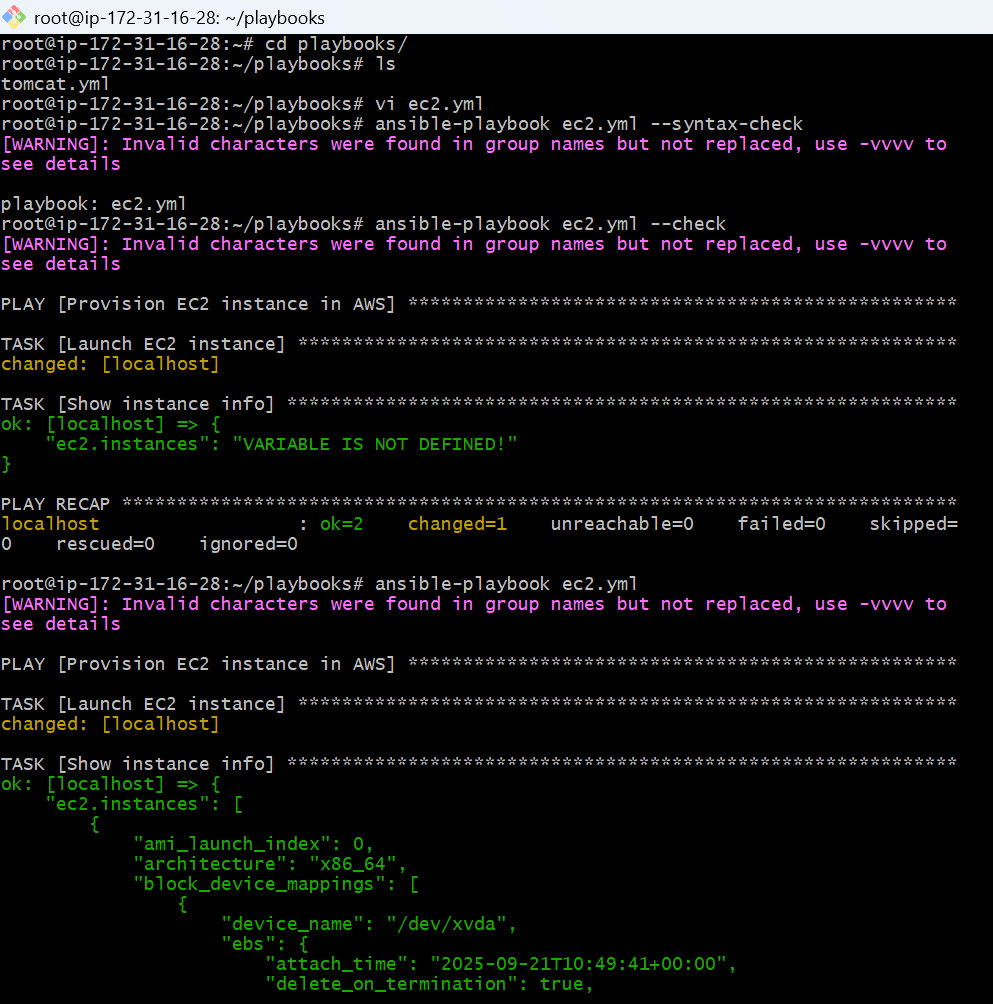
ansible-playbook ec2.yml --syntax-check

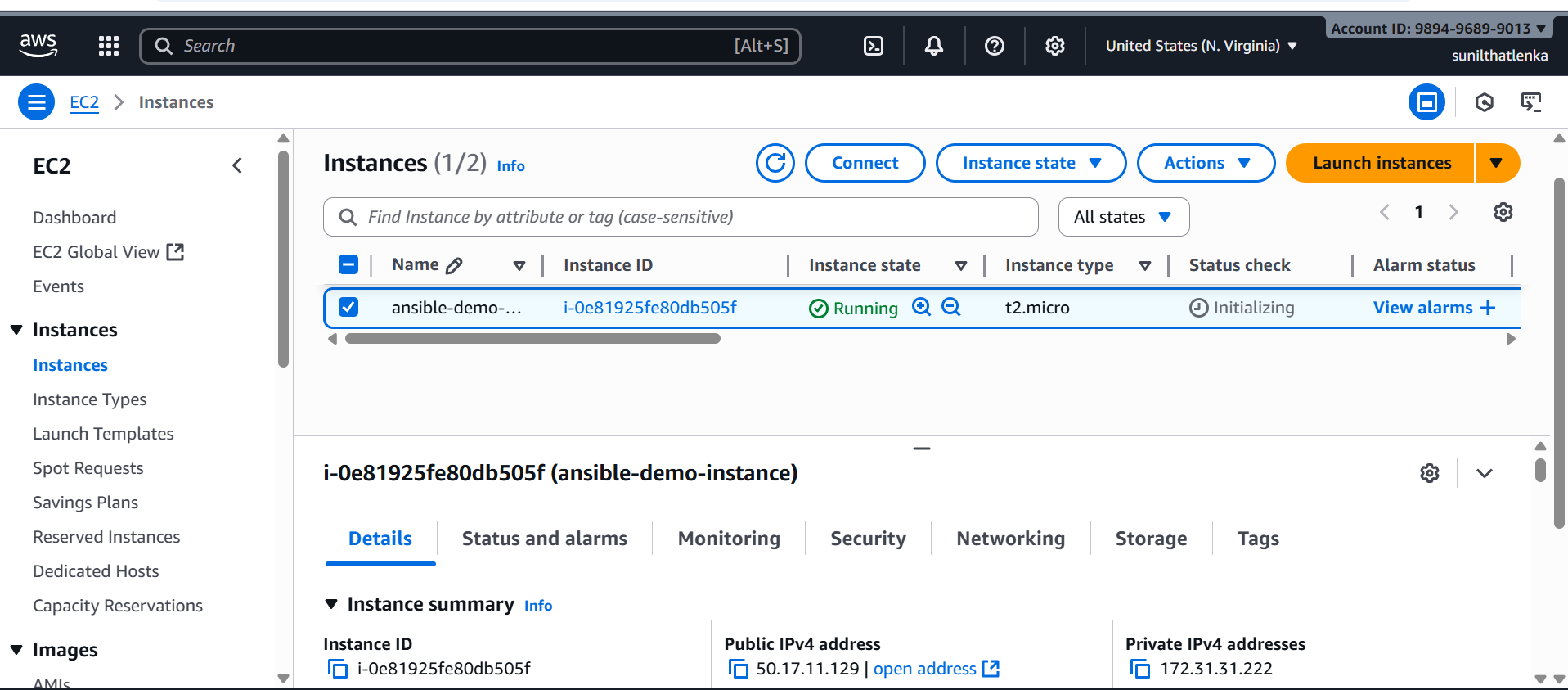
Dry-run:

ansible-playbook ec2.yml --check

Execute:

ansible-playbook ec2.yml

****

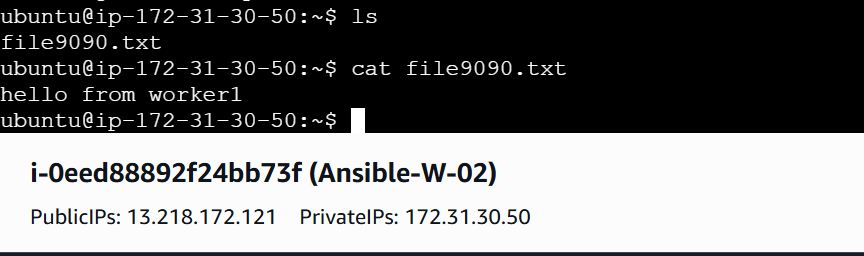
****

**5.Write a ansible playbook to copy one file from node-1 to node-2.**

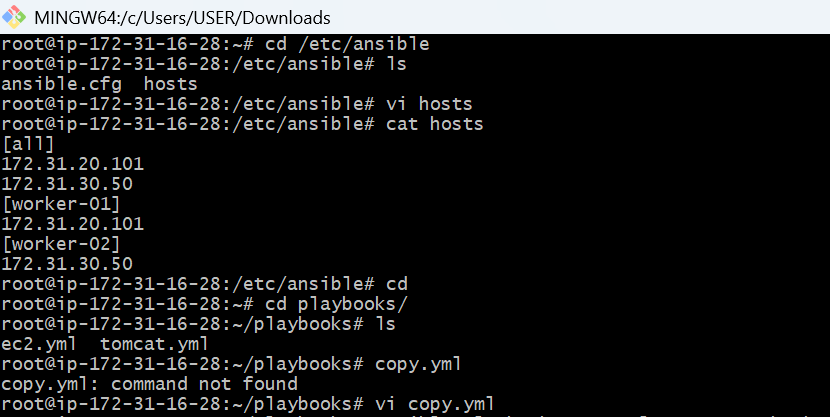
By default, Ansible’s copy module copies from the **controller → node**, not **node → node**.  
To move files **between nodes**, you need to:

1. **Fetch the file** from node-1 to the controller.
2. **Copy the file** from the controller to node-2.

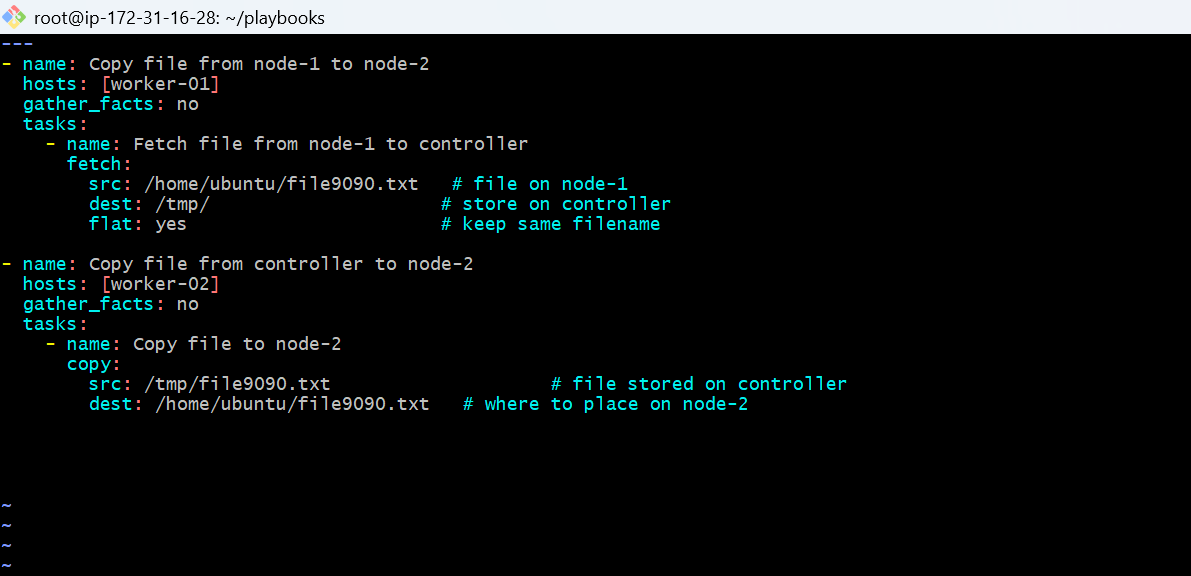
**Step -1 create a file in worker-01**

****

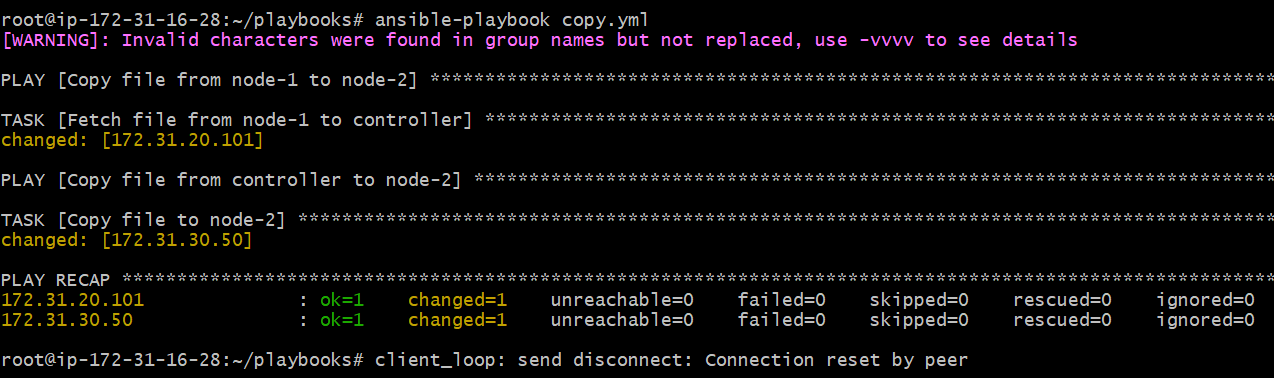
**Step -2 create copy.yml in playbook**

****

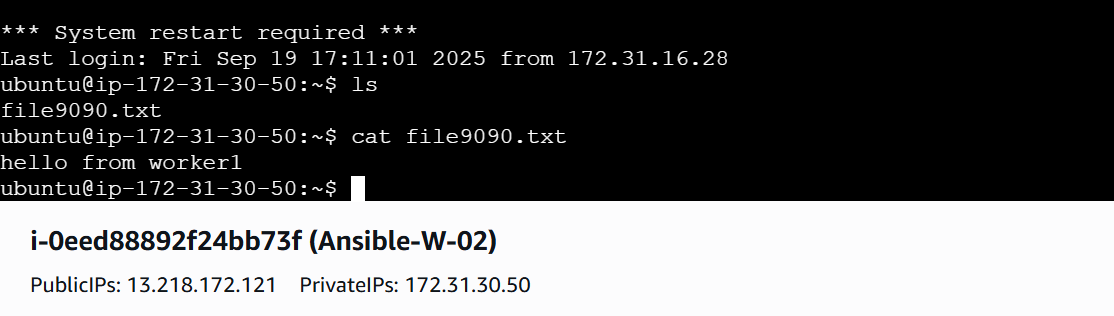
**Vi copy.yml**

****

**Run**

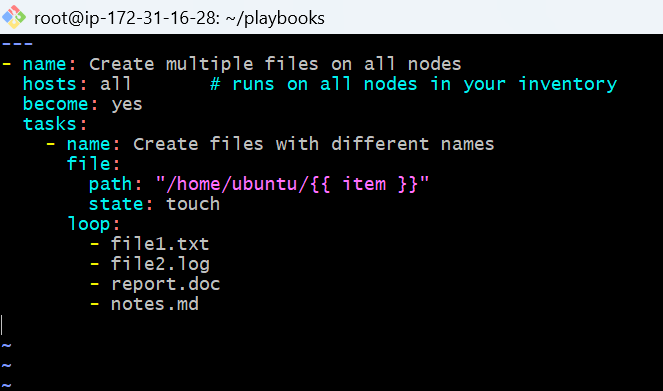
****

**Check the file in wokernode -02**

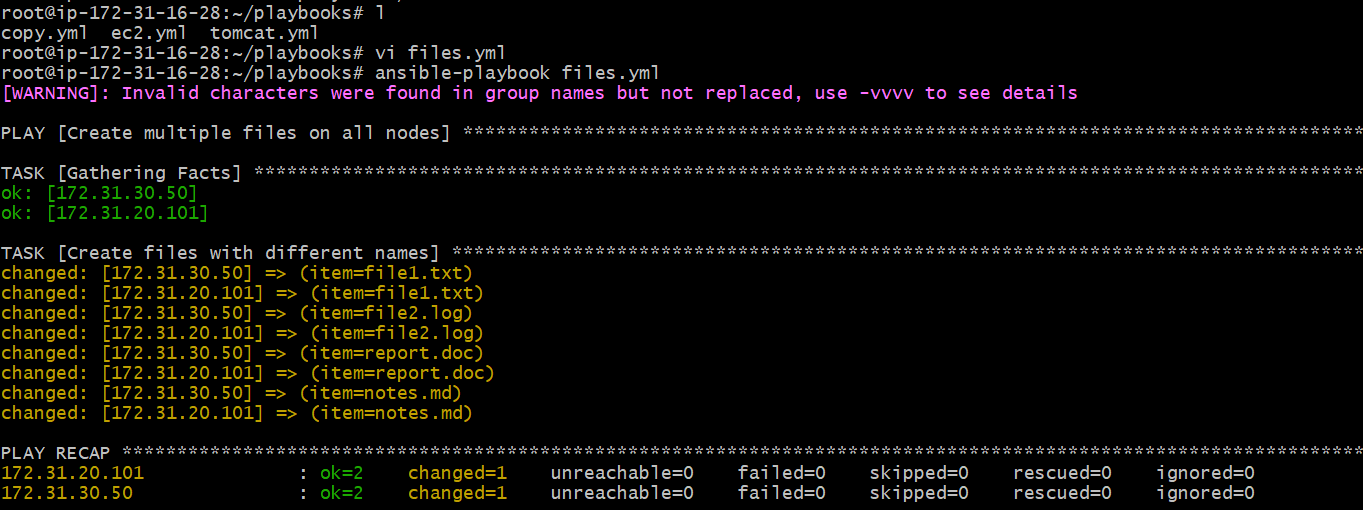
****

**6.Write a ansible playbook to create different files with different names using single playbook.**

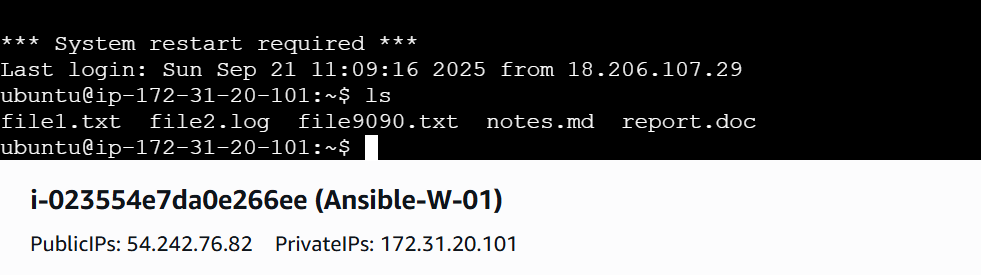
**Step -1 create a playbook**

****

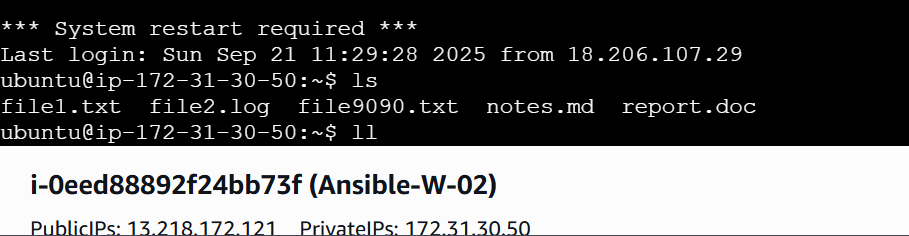
**Run**

****

**Check the files in worker nodes**

****

**W-02**

****