Step 1: Create the main list of all tasks

tasks/stop_app.yml file

```
---

- name: Gather EC2 instance metadata

  action: ec2_facts



- name: Stop application on {{ansible_hostname}}

  command: wget
"http://{{tomcat_user}}:{{tomcat_pwd}}@{{ansible_ec2_public_ipv4}}:8080/manager/text/stop?path=/HelloWorld-Maven" -O - -q
```

ii- **tasks/uninstall_app.yml**

```
---

- name: Gather EC2 instance metadata

  action: ec2_facts



- name: Undeploy application on {{ansible_hostname}}

  command: wget
"http://{{tomcat_user}}:{{tomcat_pwd}}@{{ansible_ec2_public_ipv4}}:8080/manager/text/undeploy?path=/HelloWorld-Maven" -O - -q
```

iii-**tasks/deploy_app.yml**

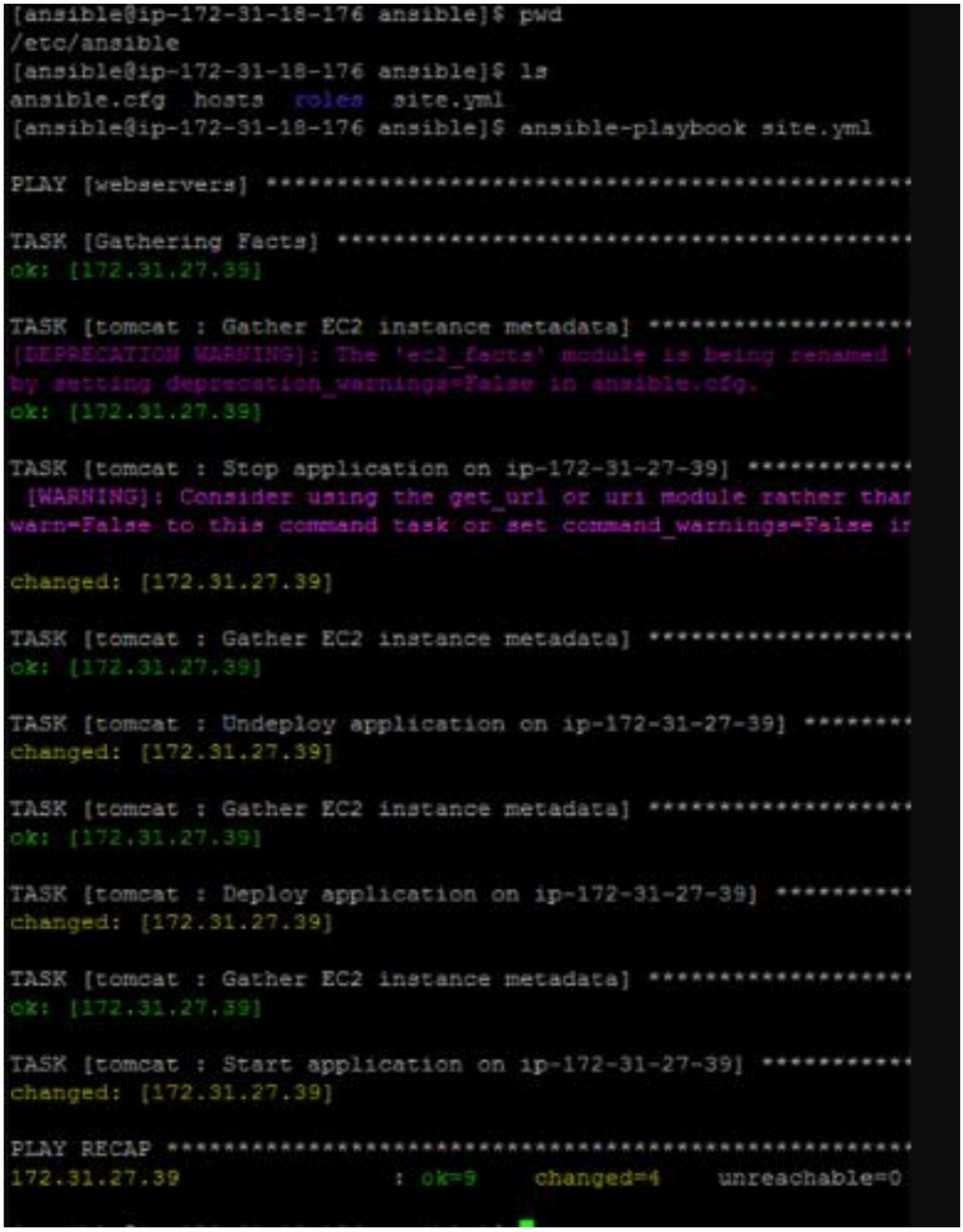
```
---

- name: Deploy the new WAR file to target servers



  copy: src=/var/lib/jenkins/workspace/Demo-Maven-Project/target/HelloWorld-
Maven.war dest=/home/ansible/tomcat/webapps
```

iv-**task/start_app.yml**

```
---

- name: Gather EC2 instance metadata

  action: ec2_facts



- name: Start application on {{ansible_hostname}}

  command: wget
"http://{{tomcat_user}}:{{tomcat_pwd}}@{{ansible_ec2_public_ipv4}}:8080/manager/text/start?path=/HelloWorld-Maven" -O - -q
```

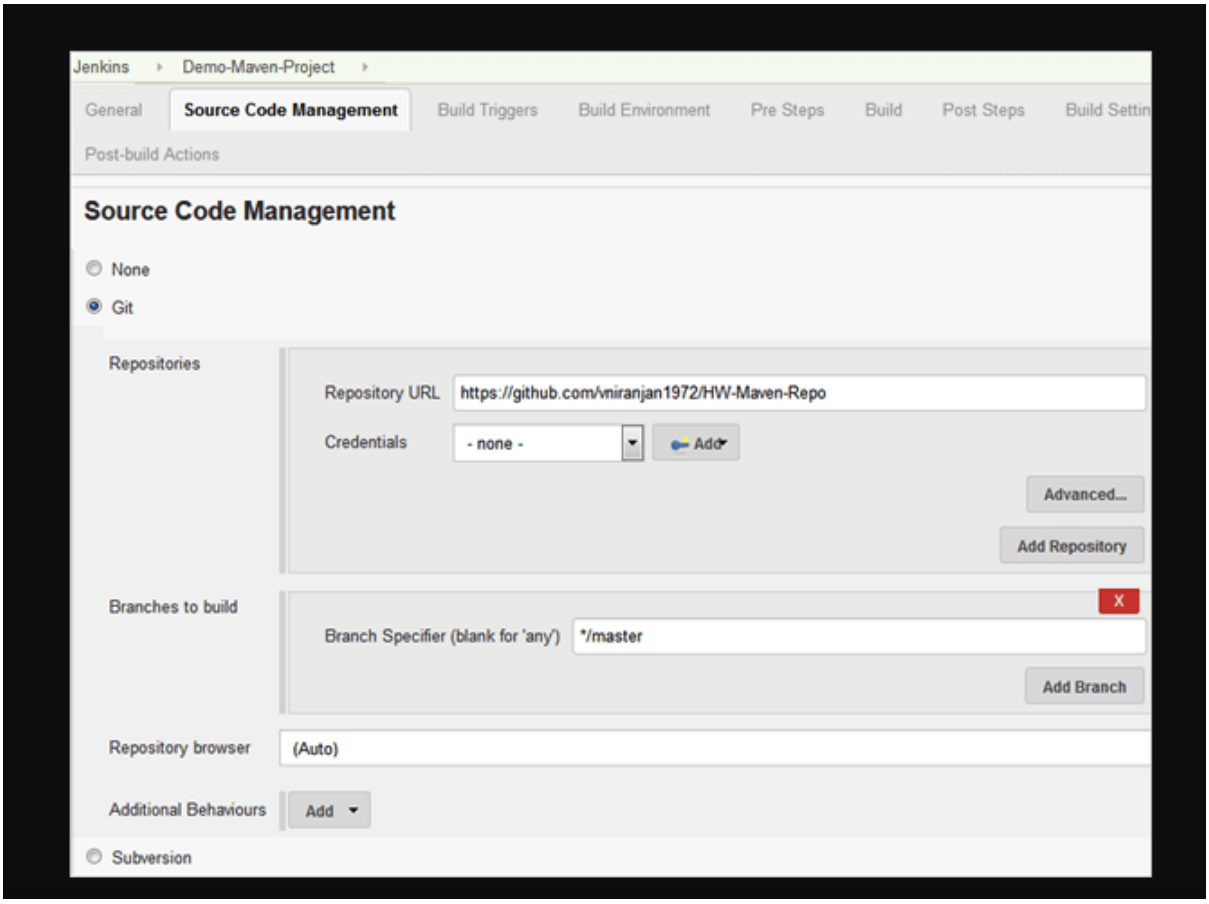
Run playbook file site.yml

- ansible-playbook site.yml



Jenkins Integration with Ansible

Step 1: Create a Jenkins job and configure the SCM repo using the code in GitHub



Configure the build

Create roles directory within the Jenkins workspace

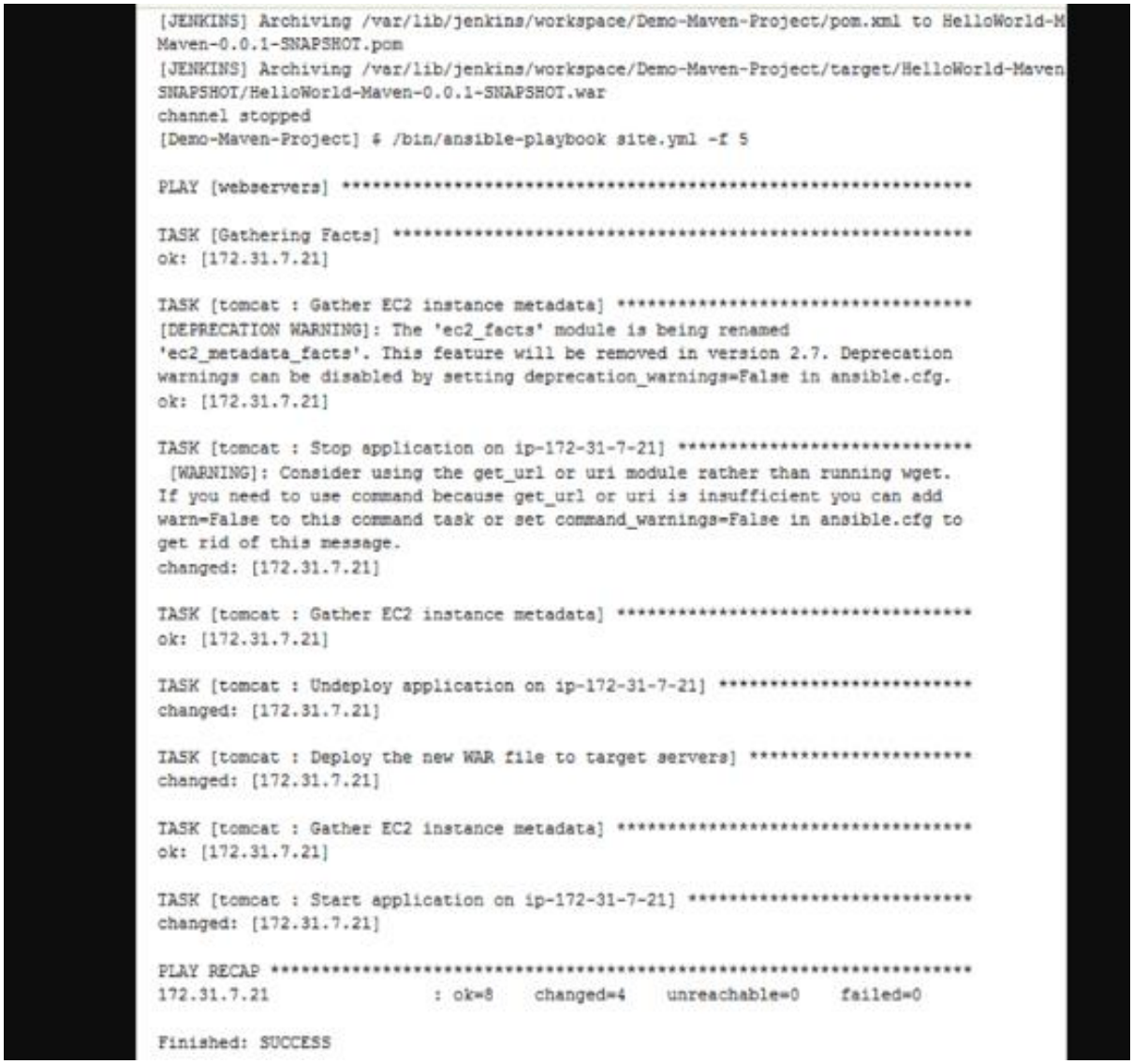
```
[ansible@ip-172-31-3-21 roles]$ pwd
/var/lib/jenkins/workspace/Demo-Maven-Project/roles
[ansible@ip-172-31-3-21 roles]$
```

Create the tomcat role in the Jenkins workspace location using the command shown below

sudo ansible-galaxy init tomcat –offline

```
[ansible@ip-172-31-3-21 roles]$ pwd
/var/lib/jenkins/workspace/Demo-Maven-Project/roles
[ansible@ip-172-31-3-21 roles]$ sudo ansible-galaxy init tomcat  --offline
- tomcat was created successfully
[ansible@ip-172-31-3-21 roles]$ ls
tomcat
[ansible@ip-172-31-3-21 roles]$ cd tomcat/
[ansible@ip-172-31-3-21 tomcat]$ tree
-bash: tree: command not found
[ansible@ip-172-31-3-21 tomcat]$ ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
```

Trigger the build job and launch the Tomcat URL to verify if the application is deployed correctly.



```
[JENKINS] Archiving /var/lib/jenkins/workspace/Demo-Maven-Project/pom.xml to HelloWorld-M
Maven-0.0.1-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/Demo-Maven-Project/target/HelloWorld-Maven
SNAPSHOT/HelloWorld-Maven-0.0.1-SNAPSHOT.war
channel stopped
[Demo-Maven-Project] $ /bin/ansible-playbook site.yml -f 5

PLAY [webservers] **************************************************************

TASK [Gathering Facts] *********************************************************
ok: [172.31.7.21]

TASK [tomcat : Gather EC2 instance metadata] ***********************************
[DEPRECATION WARNING]: The 'ec2_facts' module is being renamed
'ec2_metadata_facts'. This feature will be removed in version 2.7. Deprecation
warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [172.31.7.21]

TASK [tomcat : Stop application on ip-172-31-7-21] *****************************
 [WARNING]: Consider using the get_url or uri module rather than running wget.
If you need to use command because get_url or uri is insufficient you can add
warn=False to this command task or set command_warnings=False in ansible.cfg to
get rid of this message.
changed: [172.31.7.21]

TASK [tomcat : Gather EC2 instance metadata] ***********************************
ok: [172.31.7.21]

TASK [tomcat : Undeploy application on ip-172-31-7-21] *************************
changed: [172.31.7.21]

TASK [tomcat : Deploy the new WAR file to target servers] **********************
changed: [172.31.7.21]

TASK [tomcat : Gather EC2 instance metadata] ***********************************
ok: [172.31.7.21]

TASK [tomcat : Start application on ip-172-31-7-21] ****************************
changed: [172.31.7.21]

PLAY RECAP *********************************************************************
172.31.7.21                : ok=8    changed=4    unreachable=0    failed=0

Finished: SUCCESS
```

Manage ec2 instance with Ansible

ensure to export the user “AWS_ACCESS_KEY_ID” and “AWS_SECRET_ACCESS_KEY”.

- Create a security group
- Create key pair and the PEM file
- Create EC2 instance
- Save the EC2 instance IP address to the ansible inventory file

Code—

```
---
- hosts: localhost
  become: true
  gather_facts: False

vars:
  region: ap-south-1
  instance_type: t2.micro
  ami: ami-5b673c34 # RedHat Linux 7.5
  hosts_file: /etc/ansible/hosts

tasks:
 - name: Create security group
 ec2_group:
  aws_access_key: <AKIAZ64X6ZCI4PFK55OD>
  aws_secret_key: <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>
  name: "vniranjan"
  description: "V Niranjan Security Group"
  region: "{{ us-east-1 }}"
  rules:
    - proto: tcp
```

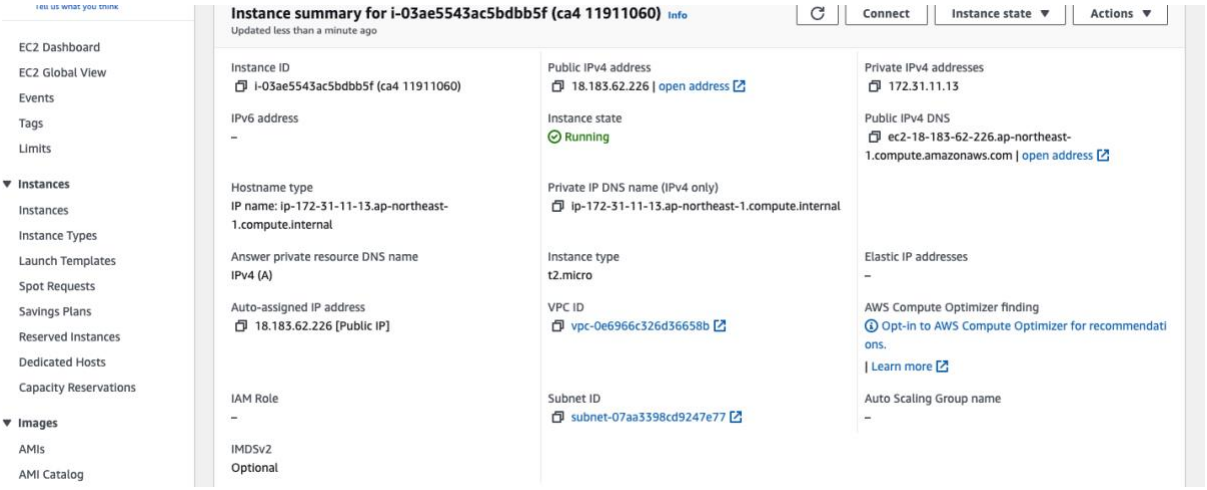
```
      from_port: 22
      to_port: 22
      cidr_ip: 0.0.0.0/0

- name: Create an EC2 key
  ec2_key:
   aws_access_key: <AKIAZ64X6ZCI4PFK55OD>
   aws_secret_key: <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>
   name: "vniranjan"
   region: "{{ us-east-1 }}"
  register: ec2_key
- name: Save private key (PEM file)
  copy: content="{{ec2_key.key.private_key}}" dest=/home/ansible/vniranjan.pem
mode=0600
  when: ec2_key.changed
- name: Create an ec2 instance
   ec2:
    aws_access_key: <AKIAZ64X6ZCI4PFK55OD>
    aws_secret_key: < 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 >
    key_name: vniranjan
    group: vniranjan # security group name
    instance_type: "{{ instance_type}}"
```

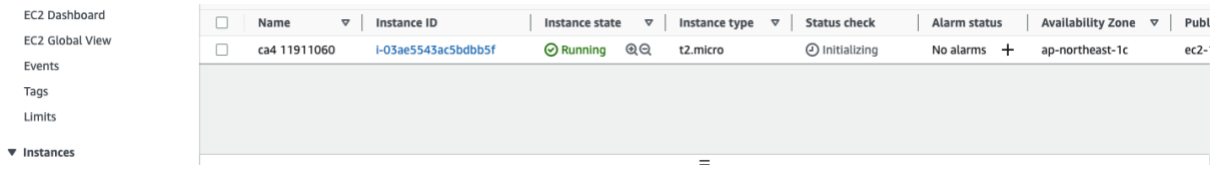
```
    image: "{{ ami }}"
    wait: true
    region: "{{ us-east-1}}"
    count: 1 # default
    count_tag:
      Name: Demo
    instance_tags:
      Name: Demo
  register: ec2
- name: Save IP to inventory file
  lineinfile:
    dest: "{{hosts_file}}"
    insertafter: '\[webservers\]'
    line: "{{item.private_ip}}"
  with_items: "{{ec2.instances}}"
```

Run the playbook

```
[ansible@ip-172-31-0-138 ~]$ ansible-playbook  createec2ins.yml

PLAY [localhost] ***************************************************************

TASK [Create security group] ***************************************************
changed: [localhost]

TASK [Create an EC2 key] *******************************************************
changed: [localhost]

TASK [Save private key (PEM file)] *********************************************
changed: [localhost]

TASK [Create an ec2 instance] **************************************************
changed: [localhost]

TASK [Save IP to inventory file] ***********************************************
changed: [localhost] => (item={u'kernel': None, u'root_device_type': u'ebs', u'p
.30.207', u'private_ip': u'172.31.12.15', u'id': u'i-0a8998c269bc8d9d9', u'ebs_o
me': u'/dev/sda1', u'ramdisk': None, u'block_device_mapping': {u'/dev/sda1': {u'
7bc'}}, u'key_name': u'vniranjan', u'image_id': u'ami-5b673c34', u'tenancy': u'd
207.ap-south-1.compute.amazonaws.com', u'state_code': 16, u'tags': {u'Name': u'D
6-30-207.ap-south-1.compute.amazonaws.com', u'region': u'ap-south-1', u'launch_t
_64', u'hypervisor': u'xen'})

PLAY RECAP *********************************************************************
localhost                  : ok=5    changed=5    unreachable=0    failed=0

[ansible@ip-172-31-0-138 ~]$
```

Ec2 instance—




EC2 Dashboard
EC2 Global View
Events
Tags
Limits
Instances
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Publ
ca4 11911060
i-03ae5543ac5bdbb5f
Running
t2.micro
Initializing
No alarms
ap-northeast-1c
ec2-