

Create A EC2 instance using Ansible

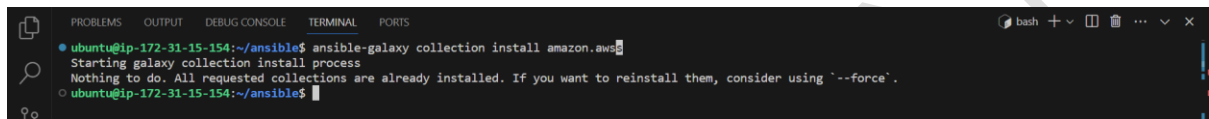
Creating an EC2 instance using Ansible involves using the `amazon.aws.ec2_instance` module, which is part of the Amazon AWS Ansible Collection. Below are the steps to accomplish this:

1. Install Required Dependencies

Install the AWS Ansible Collection

Run the following command to install the required collection:

```
ansible-galaxy collection install amazon.aws
```



```
ubuntu@ip-172-31-15-154:~/ansible$ ansible-galaxy collection install amazon.aws
Starting galaxy collection install process
Nothing to do. All requested collections are already installed. If you want to reinstall them, consider using '--force'.
```

Install boto3

```
pip install boto3
```

2. Write a Playbook For ec2_instance.

- hosts: localhost

connection: local

tasks:

- name: start an instance with a public IP address

amazon.aws.ec2_instance:

name: "ansible-instance"

key_name: "prod-ssh-key"

vpc_subnet_id: subnet-013744e41e8088axx

instance_type: t2.micro

security_group: default

region: ap-south-1

aws_access_key: "{{ec2_access_key}}" # Defined in vault

aws_secret_key: "{{ec2_secret_key}}" # Defined in Vault

network:

assign_public_ip: true

image_id: ami-053b12d3152c0cc71 #Replace the AMI_ID

```

1  ---
2  - hosts: localhost
3    connection: local
4    tasks:
5      - name: start an instance with a public IP address
6        amazon.aws.ec2_instance:
7          name: "ansible-instance"
8          # key_name: "prod-ssh-key"
9          # vpc_subnet_id: subnet-013744e41e8088axx
10         instance_type: t2.micro
11         security_group: default
12         region: ap-south-1
13         aws_access_key: "{{ec2_access_key}}" # Defined in vault
14         aws_secret_key: "{{ec2_secret_key}}" # Defined in Vault
15         network:
16           assign_public_ip: true
17         image_id: ami-053b12d3152c0cc71 #Replace the AMI_ID

```

- To Communicate with Aws we need Access key and Sceret access key.
- Generate the keys and store in ansible-vault.

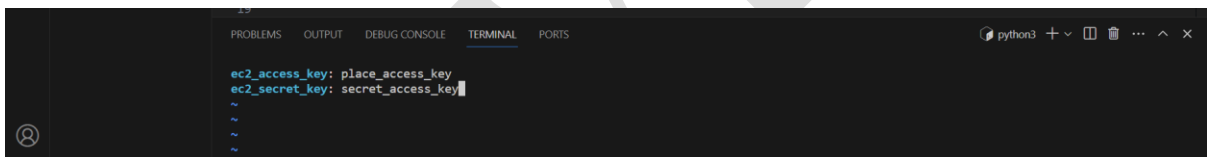
Setup Vault

1. Create a password for vault

```
openssl rand -base64 2048 > vault.pass
```

2. Add your AWS credentials using the below vault command

```
ansible-vault create group_vars/all/pass.yml --vault-password-file vault.pass
```

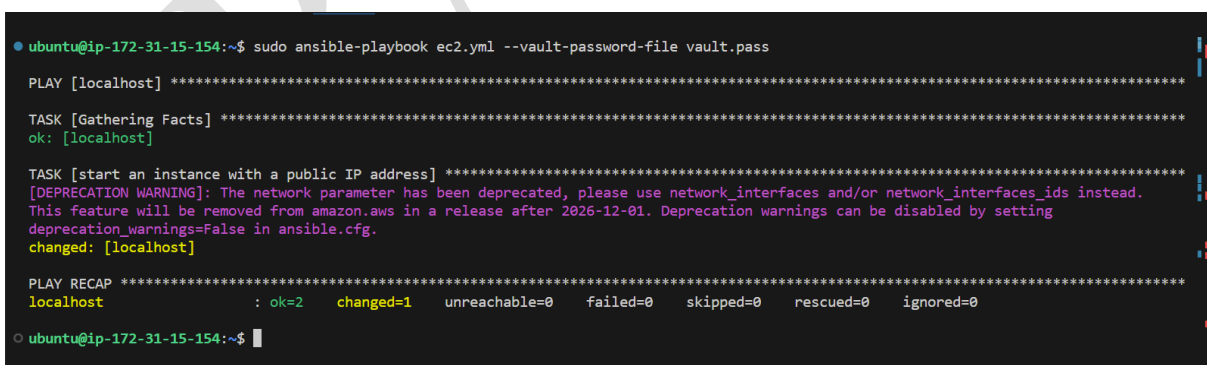


```

ec2_access_key: place_access_key
ec2_secret_key: secret_access_key
~
~
~

```

- Place the keys and save the file.
- Now Execute the playbook.



```

ubuntu@ip-172-31-15-154:~$ sudo ansible-playbook ec2.yml --vault-password-file vault.pass

PLAY [localhost] *****

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

TASK [start an instance with a public IP address] *****
[DEPRECATION WARNING]: The network parameter has been deprecated, please use network_interfaces and/or network_interfaces_ids instead.
This feature will be removed from amazon.aws in a release after 2026-12-01. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
changed: [localhost]

PLAY RECAP *****
localhost      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-15-154:~$

```

Instances							
Instance Types	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Launch Templates	ansible-instance	i-05142287216782613	Running	t2.micro	Initializing	View alarms +	ap-south-1a
Spot Requests							