

# ANSIBLE

G.Vinay

## Method-1

- ❖ Creating the AWS resources by using the playbook script.

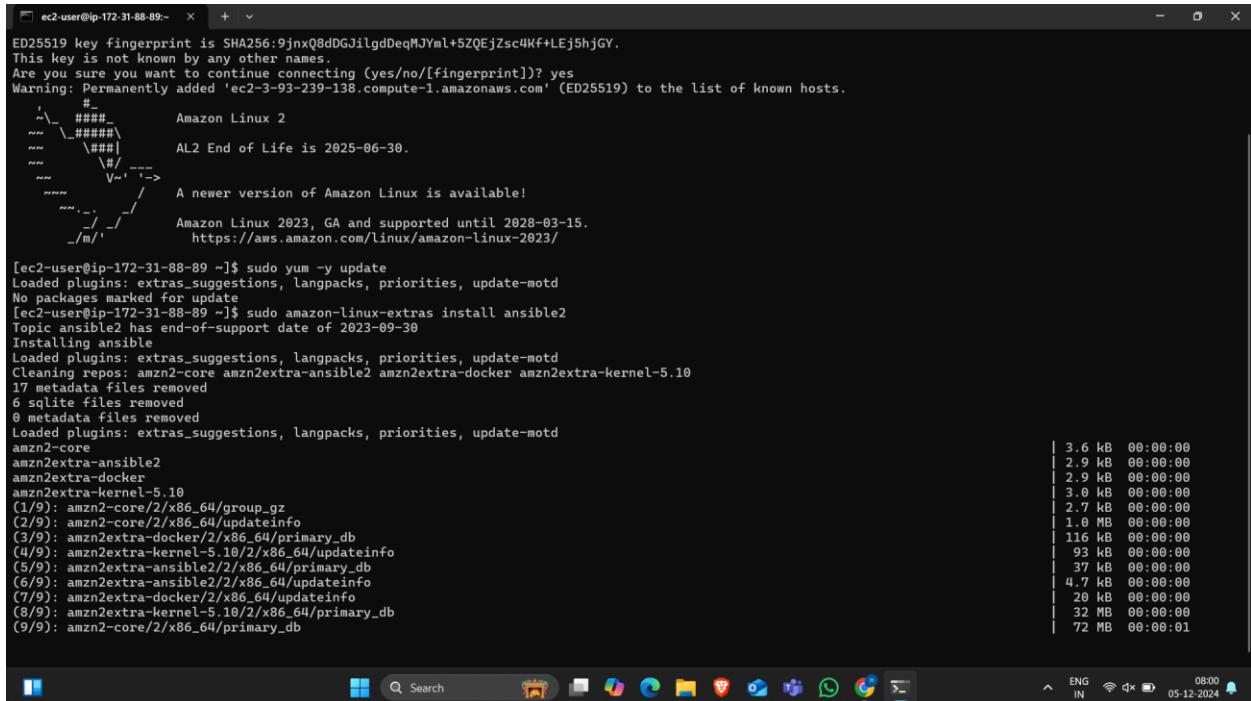
1. We have to launch an instance by using amazon-linux-2.

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named 'Ansible' is displayed with a value of 1.0. The metric has a unit of 'Count' and is associated with the 'aws.ec2.instance.state' dimension. The time range is set from 'Last 1 hour' to 'Last 1 day'. The chart shows a constant value of 1.0 across the entire period.

2. Connect to the terminal and update it.

```
ec2-user@ip-172-31-88-89:~ % cd Downloads
ec2-user@ip-172-31-88-89:~/Downloads % ssh -i "vinay.pem" ec2-user@ec2-3-93-239-138.compute-1.amazonaws.com
Last login: Thu Dec 5 02:29:48 2024 from 49.204.100.179
[ec2-user@ip-172-31-88-89 ~] $ ansible all -m ping
ansible [WARNING]: UNREACHABLE! => {
    "msg": "Failed to connect to the host via ssh: Warning: Permanently added '172.31.88.89' (ECDSA) to the list of known hosts.\r\nnno such identity: /home/ec2-user/vinay.pem: No such file or directory\r\nPermission denied (publickey,gssapi-keyex,gssapi-with-mic).",
    "unreachable": true
}
[ec2-user@ip-172-31-88-89 ~] $ ll
total 0
[ec2-user@ip-172-31-88-89 ~] $ chmod 400 vinay.pem
[ec2-user@ip-172-31-88-89 ~] $ sudo chmod 600 vinay.pem
[ec2-user@ip-172-31-88-89 ~] $ ansible all -m ping
ansible [WARNING]: SUCCESS => {
    "changed": false,
    "ping": "pong"
}
[ec2-user@ip-172-31-88-89 ~] $ |
```

### 3. Install the ansible “sudo amazon-linux-extras install ansible2”.



```
ED25519 key fingerprint is SHA256:9jnxQ8dDGJilgdDeqMJYml+5ZQEjZsc4Kf+LEj5hjGY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-93-239-138.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

Amazon Linux 2
AL2 End of Life is 2025-06-30.

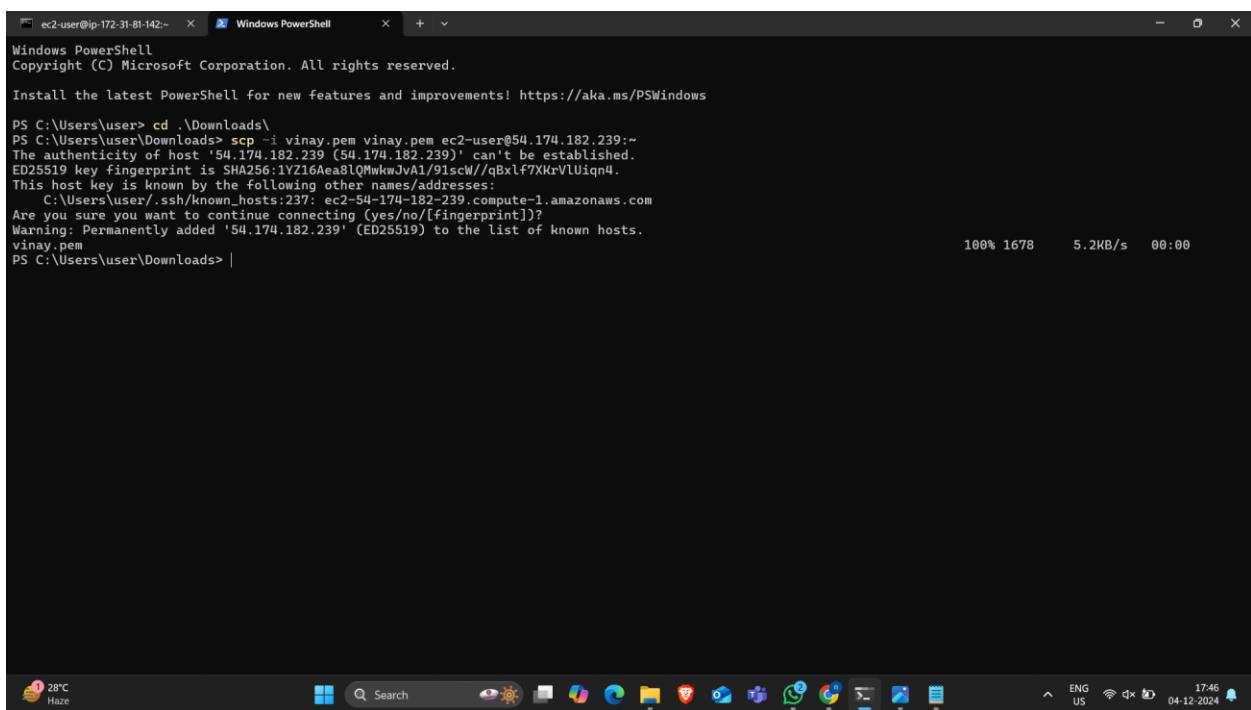
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-88-89 ~]$ sudo yum -y update
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-172-31-88-89 ~]$ sudo amazon-linux-extras install ansible2
Topic ansible2 has end-of-support date of 2023-09-30
Installing ansible
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-ansible2 amzn2extra-docker amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-ansible2
amzn2extra-docker
amzn2extra-kernel-5.10
(1/9): amzn2-core/2/x86_64/group.gz 3.6 kB 00:00:00
(2/9): amzn2-core/2/x86_64/updateinfo 2.9 kB 00:00:00
(3/9): amzn2extra-docker/2/x86_64/primary_db 2.9 kB 00:00:00
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo 3.0 kB 00:00:00
(5/9): amzn2extra-ansible2/2/x86_64/primary_db 2.7 kB 00:00:00
(6/9): amzn2extra-ansible2/2/x86_64/updateinfo 1.0 MB 00:00:00
(7/9): amzn2extra-docker/2/x86_64/updateinfo 116 kB 00:00:00
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db 93 kB 00:00:00
(9/9): amzn2-core/2/x86_64/primary_db 37 kB 00:00:00
4.7 kB 00:00:00
20 kB 00:00:00
32 MB 00:00:00
72 MB 00:00:01

[ec2-user@ip-172-31-88-89 ~]$
```

4. Copy the pem file from the device to the local server “scp -i pem pem ec2-user@pub\_ip:~”

5. Give permission for pem file.



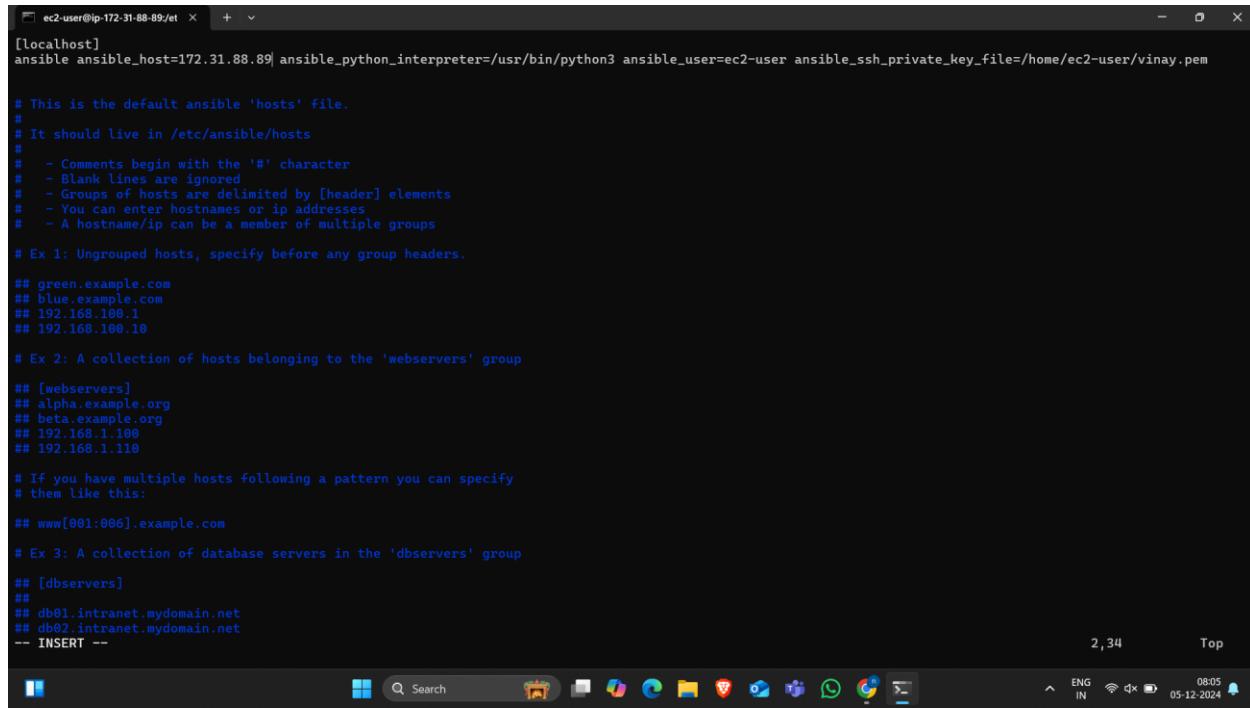
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\user> cd ..\Downloads\
PS C:\Users\user\Downloads> scp -i vinay.pem vinay.pem ec2-user@54.174.182.239:~
The authenticity of host '54.174.182.239 (54.174.182.239)' can't be established.
ED25519 key fingerprint is SHA256:YZ16Aea8lQMuwuJwA1/91scW//qBxL7XKrVLUiqn4.
This host key is known by the following other names/addresses:
C:\Users\user\.ssh\known_hosts:237: ec2-54-174-182-239.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added '54.174.182.239' (ED25519) to the list of known hosts.
vinay.pem
100% 1678      5.2KB/s   00:00
PS C:\Users\user\Downloads> |
```

6. Edit the host file as shown in below.

```
cd /etc/ansible/
```



```
[ec2-user@ip-172-31-88-89:~] ansible ansible_host=172.31.88.89 ansible_python_interpreter=/usr/bin/python3 ansible_user=ec2-user ansible_ssh_private_key_file=/home/ec2-user/vinay.pem

# This is the default ansible 'hosts' file.
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.118

# If you have multiple hosts following a pattern you can specify
# them like this:

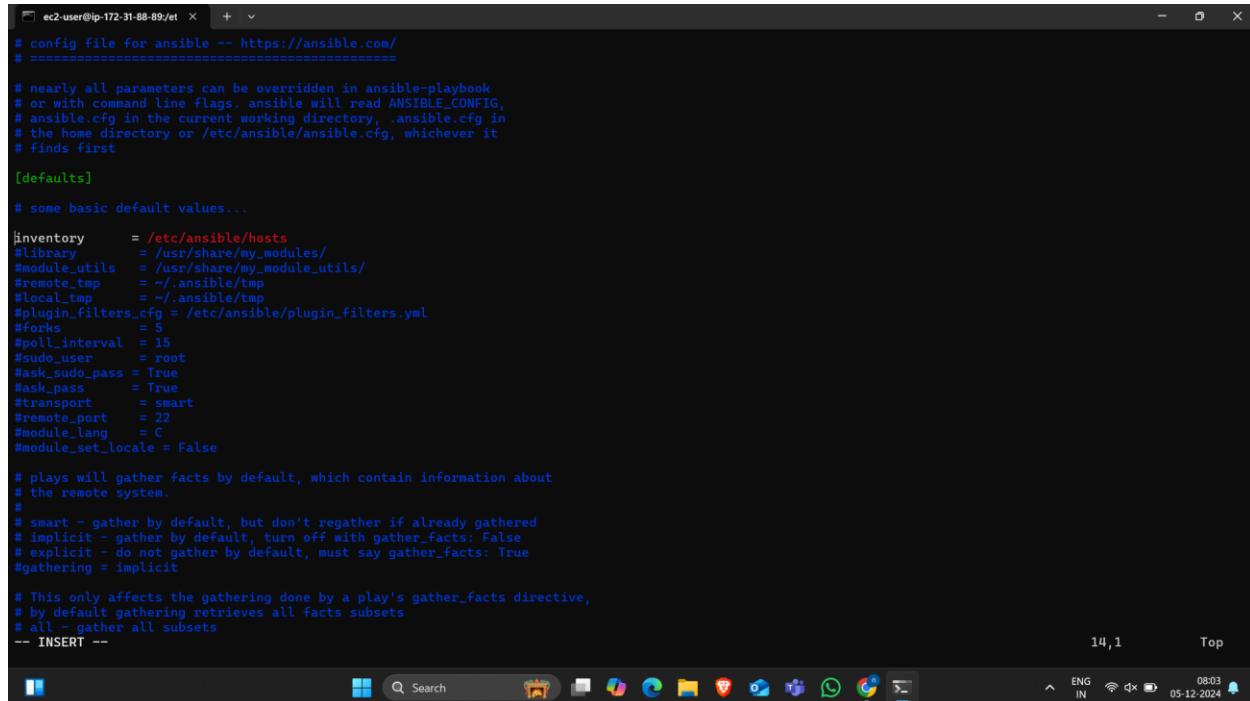
## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
-- INSERT --
```

7. Edit the configure file as shown in below.

```
cd /etc/ansible/
```



```
# config file for ansible -- https://ansible.com/
# =====

# nearly all parameters can be overridden in ansible-playbook
# or with command line flags. ansible will read ANSIBLE_CONFIG,
# ansible.cfg in the current working directory, .ansible.cfg in
# the home directory or /etc/ansible/ansible.cfg, whichever it
# finds first

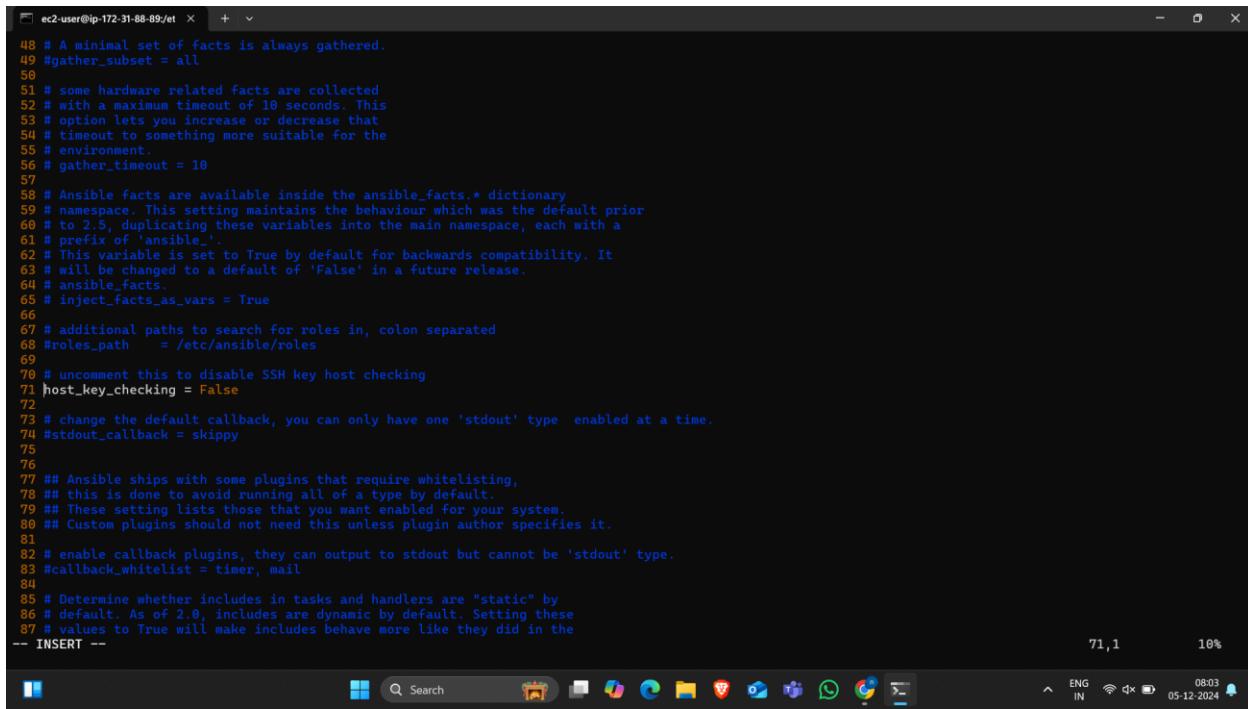
[defaults]

# some basic default values...

inventory      = /etc/ansible/hosts
#library        = /usr/share/my_modules/
#module_utils   = /usr/share/my_module_utils/
#remote_tmp     = ~/.ansible/tmp
#local_tmp      = ~/.ansible/tmp
#plugin_filters_cfg = /etc/ansible/plugin_filters.yml
#forks          = 5
#poll_interval  = 15
#sudo_user      = root
#ask_sudo_pass  = True
#ask_pass        = True
#transport      = smart
#remote_port    = 22
#module_lang    = C
#module_set_locale = False

# plays will gather facts by default, which contain information about
# the remote system.
#
# smart - gather by default, but don't regather if already gathered
# implicit - gather by default, turn off with gather_facts: False
# explicit - do not gather by default, must say gather_facts: True
#gathering = implicit

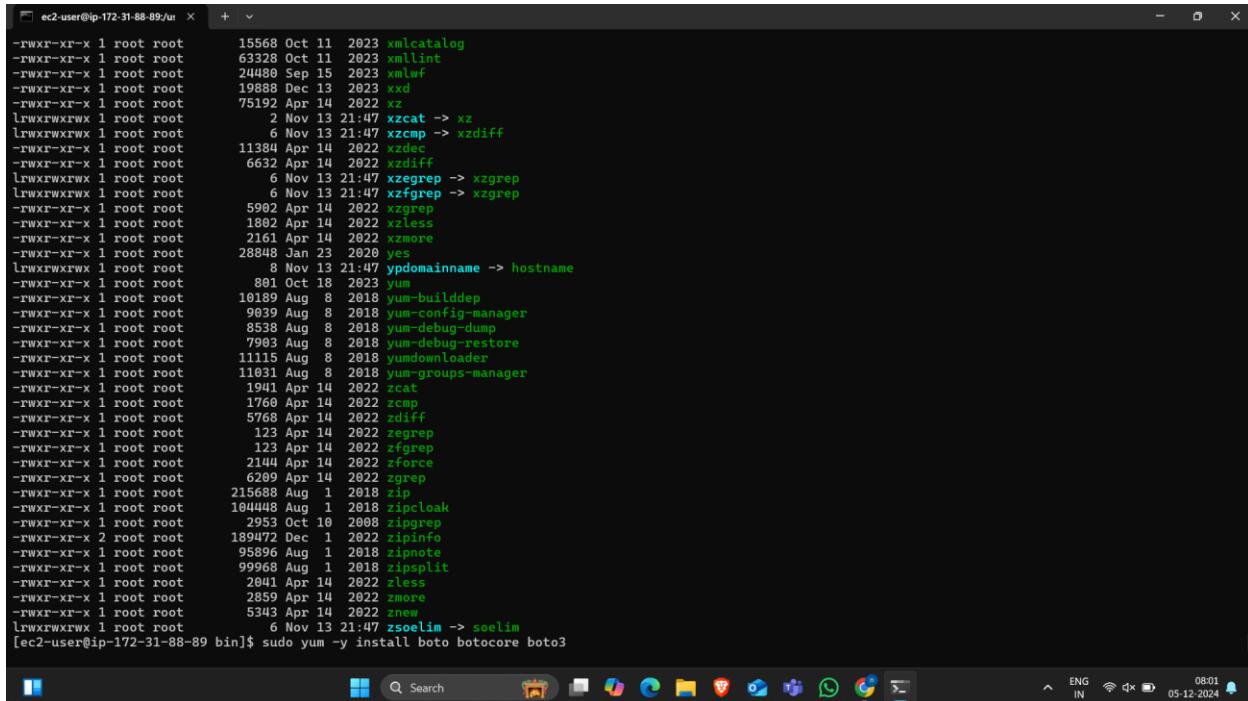
# This only affects the gathering done by a play's gather_facts directive,
# by default gathering retrieves all facts subsets
# all - gather all subsets
-- INSERT --
```



```
ec2-user@ip-172-31-88-89:et ~ + \n\n48 # A minimal set of facts is always gathered.\n49 #gather_subset = all\n50\n51 # some hardware related facts are collected\n52 # with a maximum timeout of 10 seconds. This\n53 # option lets you increase or decrease that\n54 # timeout to something more suitable for the\n55 # environment.\n56 # gather_timeout = 10\n57\n58 # Ansible facts are available inside the ansible_facts.* dictionary\n59 ## namespace. This setting maintains the behaviour which was the default prior\n60 ## to 2.5, duplicating these variables into the main namespace, each with a\n61 ## prefix of 'ansible_'.\n62 # This variable is set to True by default for backwards compatibility. It\n63 # will be changed to a default of 'False' in a future release.\n64 # ansible_facts.\n65 # inject_facts_as_vars = True\n66\n67 # additional paths to search for roles in, colon separated\n68 #roles_path      = /etc/ansible/roles\n69\n70 # uncomment this to disable SSH key host checking\n71 host_key_checking = False\n72\n73 # change the default callback, you can only have one 'stdout' type enabled at a time.\n74 #stdout_callback = skippy\n75\n76\n77 ## Ansible ships with some plugins that require whitelisting,\n78 ## this is done to avoid running all of a type by default.\n79 ## These setting lists those that you want enabled for your system.\n80 ## Custom plugins should not need this unless plugin author specifies it.\n81\n82 # enable callback plugins, they can output to stdout but cannot be 'stdout' type.\n83 #callback_whitelist = timer, mail\n84\n85 # Determine whether includes in tasks and handlers are "static" by\n86 # default. As of 2.0, includes are dynamic by default. Setting these\n87 # values to True will make includes behave more like they did in the\n-- INSERT --
```

8. Go to path /usr/bin/python3 and install boto boto3 botocore.

“sudo pip3 install boto botocore boto3”



```
ec2-user@ip-172-31-88-89:ui ~ + \n\n15568 Oct 11 2023 xmllint\n63328 Oct 11 2023 xmllint\n244880 Sep 15 2023 xmlfw\n19888 Dec 13 2023 xxd\n75192 Apr 14 2022 xz\n2 Nov 13 21:47 xzcat -> xz\n6 Nov 13 21:47 xzcmp -> xzdiff\n11384 Apr 14 2022 xzdec\n6632 Apr 14 2022 xzdiff\n6 Nov 13 21:47 xzegrep -> xzgrep\n6 Nov 13 21:47 xzfgrep -> xzgrep\n5982 Apr 14 2022 xzgrep\n1882 Apr 14 2022 xzless\n2161 Apr 14 2022 xzmore\n28848 Jan 23 2026 yes\n8 Nov 13 21:47 ypdomainname -> hostname\n891 Oct 18 2023 yum\n10189 Aug 8 2018 yum-builddep\n9039 Aug 8 2018 yum-config-manager\n8538 Aug 8 2018 yum-debug-dump\n7993 Aug 8 2018 yum-debug-restore\n11115 Aug 8 2018 yumdownloader\n11831 Aug 8 2018 yum-groups-manager\n1941 Apr 14 2022 zcat\n1768 Apr 14 2022 zcmp\n5768 Apr 14 2022 zdif\n123 Apr 14 2022 zgrep\n123 Apr 14 2022 zfgrep\n2144 Apr 14 2022 zforce\n6299 Apr 14 2022 zgrep\n215688 Aug 1 2018 zip\n104448 Aug 1 2018 zipcloak\n2953 Oct 10 2008 zipgrep\n189472 Dec 1 2022 zipinfo\n95896 Aug 1 2018 zipnote\n99968 Aug 1 2018 zipsplit\n2041 Apr 14 2022 zless\n2859 Apr 14 2022 zmore\n5343 Apr 14 2022 znew\n6 Nov 13 21:47 zsoelim -> soelim\n[ec2-user@ip-172-31-88-89 bin]$ sudo yum -y install boto botocore boto3
```

9. Create the file “name.yaml” and write the yaml script for the creating resources.  
As shown in below pictures.

```

hosts: localhost
become: yes
tasks:
- ec2_vpc_net:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    cidr_block: 10.0.0.0/16
    name: vinay-vpc
    region: us-east-1
    state: present
    register: vpc_result

- ec2_vpc_igw:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    state: present
    tags:
      Name: Vinay-igw
    register: igw_result

- ec2_vpc_subnet:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    az: us-east-1a
    state: present
    cidr: 10.0.0.0/20
    map_public: yes
    resource_tags:
      Name: vinay-pub
    register: pubsubnet_result

- ec2_vpc_subnet:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    az: us-east-1b
    state: present
    cidr: 10.0.16.0/20
    map_public: no
    tags:
      Name: vinay-pvt
    register: pvtsubnet_result

- ec2_vpc_route_table:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    state: present
    tags:
      Name: vinay-pub
    subnets: [ "{{ pubsubnet_result.subnet.id }} " ]
    routes:
      - dest: 0.0.0.0/0
        gateway_id: "{{ igw_result.gateway_id }}"
    register: public_route_table

- ec2_vpc_route_table:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    state: present
    tags:

```

1,3 Top

SA - SL In 2 hours

Search

11:25 05-12-2024

```

- ec2_vpc_subnet:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    az: us-east-1b
    state: present
    cidr: 10.0.16.0/20
    map_public: no
    tags:
      Name: vinay-pvt
    register: pvtsubnet_result

- ec2_vpc_route_table:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    state: present
    tags:
      Name: vinay-pub
    subnets: [ "{{ pvtsubnet_result.subnet.id }} " ]
    routes:
      - dest: 0.0.0.0/0
        gateway_id: "{{ public_route_table.gateway_id }}"
    register: public_route_table

- ec2_vpc_route_table:
    aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
    aws_secret_key: "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"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: us-east-1
    state: present
    tags:

```

72, 49 33%

SA - SL In 2 hours

Search

11:26 05-12-2024

```
ec2-user@ip-172-31-88-89: ~ % Windows PowerShell ~ % Windows PowerShell ~ % + - 
aws_secret_key: "fh9F8UP1fZIu9zdx0DQczJXMIGWFN9y+nik5E5Gk"
vpc_id: "{{ vpc_result.vpc.id }}"

region: us-east-1
state: present
tags:
  Name: vinay-pvt-rt
subnets: [{"vpcsubnet_result.subnet.id"}]
register: private_route_table

- ec2_group:
  aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
  aws_secret_key: "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"
  vpc_id: "{{ vpc_result.vpc.id }}"
  region: us-east-1
  state: present
  name: vinay-sg
  description: allow
  tags:
    Name: vinnu-sg
  rules:
    - proto: all
      cidr_ip: 0.0.0.0/0
      rule_desc: allow all traffic
register: security_group_results

- ec2:
  image: ami-0166fe664262f664c
  instance_type: t2.micro
  region: us-east-1
  wait: yes
  count: 1
  state: present
  vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}"
  assign_public_ip: yes
  group_id: "{{ security_group_results.group_id }}"
  aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
  aws_secret_key: "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"
  user_data: "[{ lookup('file', 'static.sh') }]"
  instance_tags:
    Name: vinu-Ec

106,29 64%
SA - SL In 2 hours 11:26 05-12-2024 ENG IN
```

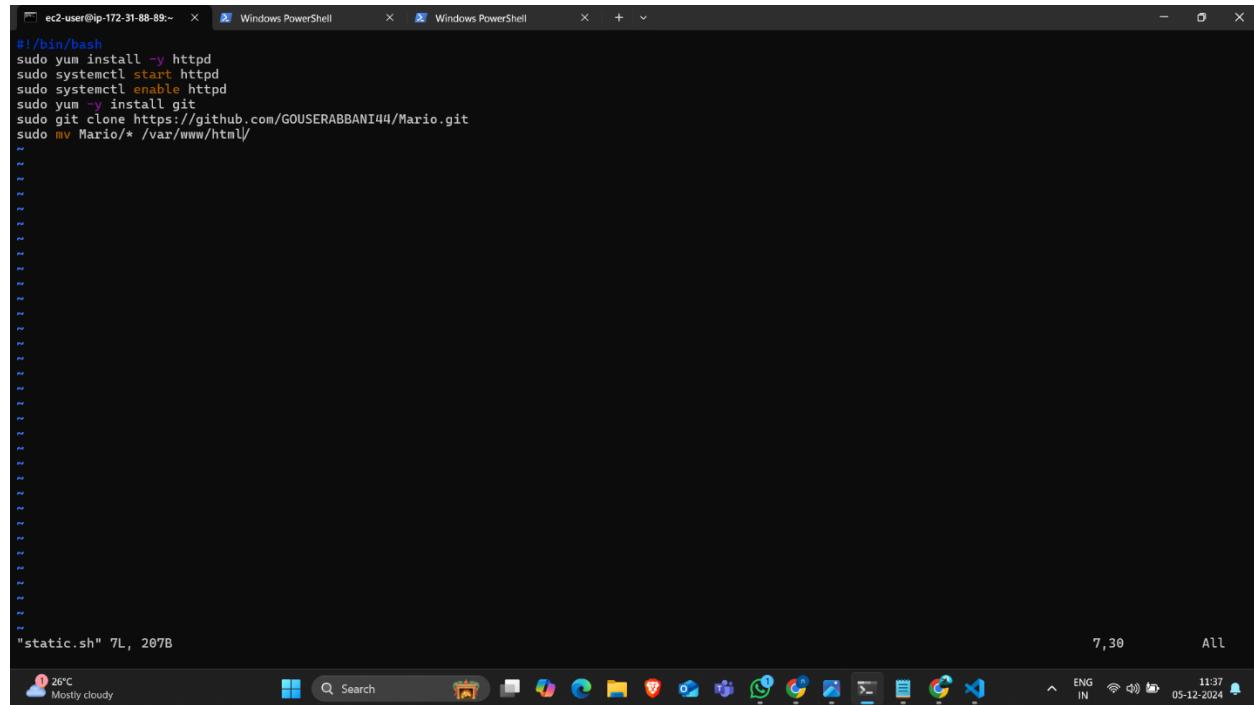
```
ec2-user@ip-172-31-88-89: ~ % Windows PowerShell ~ % Windows PowerShell ~ % + - 
cidr_ip: 0.0.0.0/0
rule_desc: allow all traffic
register: security_group_results

- ec2:
  image: ami-0166fe664262f664c
  instance_type: t2.micro
  region: us-east-1
  wait: yes
  count: 1
  state: present
  vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}"
  assign_public_ip: yes
  group_id: "{{ security_group_results.group_id }}"
  aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
  aws_secret_key: "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"
  user_data: "[{ lookup('file', 'static.sh') }]"
  instance_tags:
    Name: vinay-Ec

- ec2:
  image: ami-0166fe664262f664c
  instance_type: t2.micro
  region: us-east-1
  wait: yes
  count: 1
  state: present
  vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}"
  assign_public_ip: yes
  group_id: "{{ security_group_results.group_id }}"
  aws_access_key: "AKIAVY2PG5RDO5KUAWCJ"
  aws_secret_key: "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"
  user_data: "[{ lookup('file', 'dynamic.sh') }]"
  instance_tags:
    Name: vinnu-Ec

128,0-1 84%
SA - SL In 2 hours 11:26 05-12-2024 ENG IN
```

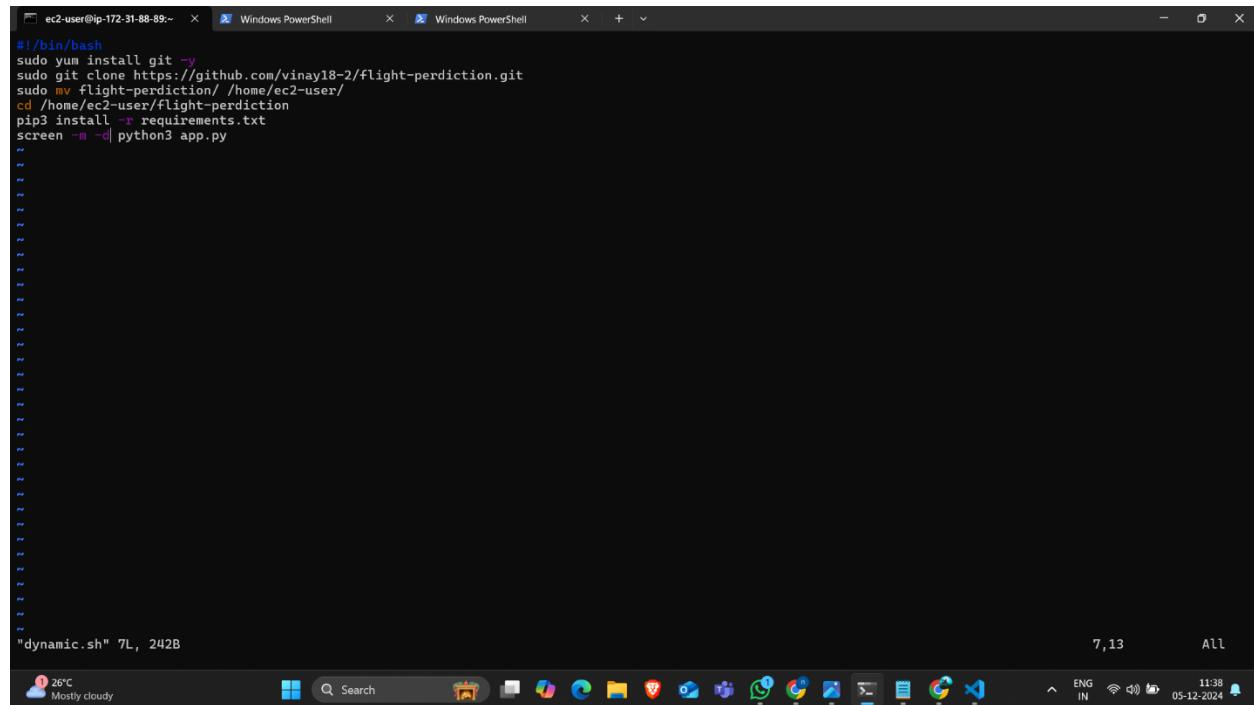
10. Create the userdata file “userdata1.sh” to host an application. Here this is a static webapplication script.



```
#!/bin/bash
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
sudo yum -y install git
sudo git clone https://github.com/Gouserabbani44/Mario.git
sudo mv Mario/* /var/www/html/
```

"static.sh" 7L, 207B

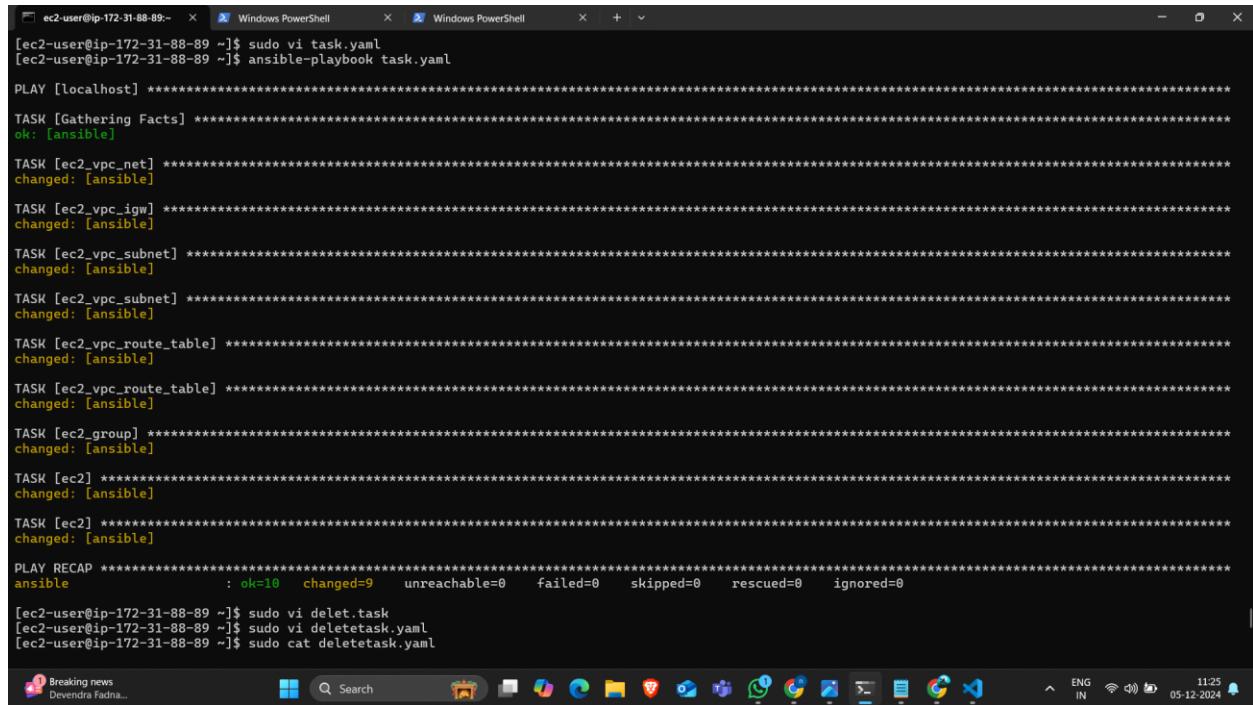
11. This the user data to host dynamic application.



```
#!/bin/bash
sudo yum install git -y
sudo git clone https://github.com/vinayl8-2/flight-perdition.git
sudo mv flight-perdition/ /home/ec2-user/
cd /home/ec2-user/flight-perdition
pip3 install -r requirements.txt
screen -m -d python3 app.py
```

"dynamic.sh" 7L, 242B

12. After writing the script than save it and execute that playbook by using the command “ansible-playbook task.yaml”
13. We can observe that the playbook has been created successfully.



```
[ec2-user@ip-172-31-88-89 ~]$ sudo vi task.yaml
[ec2-user@ip-172-31-88-89 ~]$ ansible-playbook task.yaml

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

TASK [ec2_vpc_net] ****
changed: [ansible]

TASK [ec2_vpc_igw] ****
changed: [ansible]

TASK [ec2_vpc_subnet] ****
changed: [ansible]

TASK [ec2_vpc_subnet] ****
changed: [ansible]

TASK [ec2_vpc_route_table] ****
changed: [ansible]

TASK [ec2_vpc_route_table] ****
changed: [ansible]

TASK [ec2_group] ****
changed: [ansible]

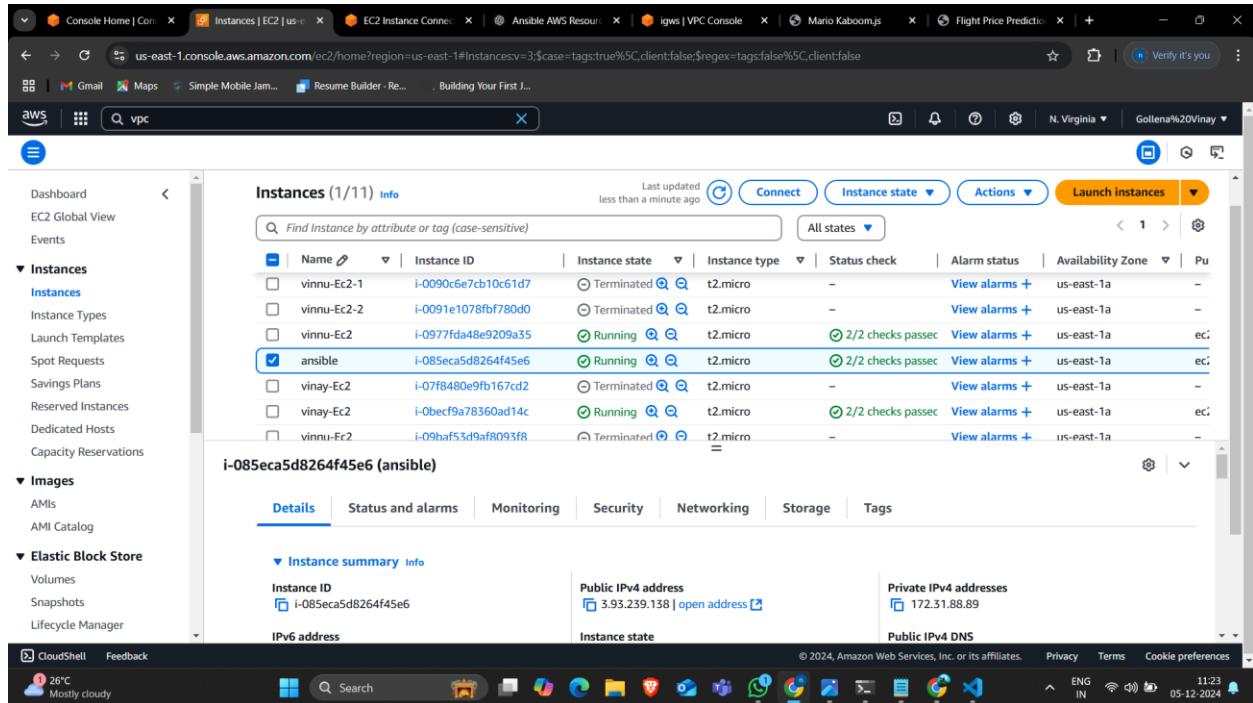
TASK [ec2] ****
changed: [ansible]

TASK [ec2] ****
changed: [ansible]

PLAY RECAP ****
ansible : ok=10    changed=9     unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-88-89 ~]$ sudo vi delet.task
[ec2-user@ip-172-31-88-89 ~]$ sudo vi deletetask.yaml
[ec2-user@ip-172-31-88-89 ~]$ sudo cat deletetask.yaml
```

14. Here the result of created instances by using playbook script.



**Instances (1/11) Info**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
vinnu-Ec2-1	i-0090c6e7cb10c61d7	Terminated	t2.micro	-	View alarms +	us-east-1a	-
vinnu-Ec2-2	i-0091e1078fbf780d0	Terminated	t2.micro	-	View alarms +	us-east-1a	-
vinnu-Ec2	i-0977fda48e9209a35	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
<b>ansible</b>	<b>i-085eca5d8264f45e6</b>	<b>Running</b>	<b>t2.micro</b>	<b>2/2 checks passed</b>	<b>View alarms +</b>	<b>us-east-1a</b>	<b>ec2</b>
vinay-Ec2	i-07ff8480e9fb167cd2	Terminated	t2.micro	-	View alarms +	us-east-1a	-
vinay-Ec2	i-0becf9a78360ad14c	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
vinnu-Ec2	i-09baf53d9af8093f8	Terminated	t2.micro	-	View alarms +	us-east-1a	-

**i-085eca5d8264f45e6 (ansible)**

**Details**    Status and alarms    Monitoring    Security    Networking    Storage    Tags

**Instance summary info**

Instance ID i-085eca5d8264f45e6	Public IPv4 address 3.95.239.138   open address	Private IPv4 addresses 172.31.88.89
IPv6 address	Instance state	Public IPv4 DNS

15. Here one instance I have hosted the static application. Copy the public ip of created ec2 instance and search in the google with port number 80.

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like Dashboard, EC2 Global View, Events, Instances (with sub-options Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots, Lifecycle Manager). The main area displays a table of 11 instances. One instance, 'vinay-Ec2' (Instance ID: i-0becf9a78360ad14c), is selected and highlighted with a blue border. The instance details show it is running, has a Public IPv4 address of 3.83.141.87, and a Private IP4 DNS of ec2-3-83-141-87.compute-1.amazonaws.com. Other instances listed include 'i-07f8480e9fb167cd2' (terminated), 'i-09ba5f53d9af8093f8' (terminated), and 'i-017cda4644283abe9' (terminated).

16. Here the result of hosted static application Mario.

The screenshot shows a web browser displaying a Mario game level. The URL bar indicates 'Not secure 3.83.141.87'. The page content is a Mario level with pipes and coins. The browser interface includes a search bar, a taskbar with various icons, and a system tray at the bottom right showing the date and time as 05-12-2024.

17. In the another instance I have hosted the dynamic application.

The screenshot shows the AWS EC2 Instances page with 11 instances listed. One instance, 'vinnu-Ec2' (i-0977fda48e9209a35), is selected and highlighted with a blue border. This instance is running, has a t2.micro instance type, and is located in the us-east-1a availability zone. Its public IPv4 address is 100.26.195.53 and its private IP DNS name is ec2-100-26-195-53.compute-1.amazonaws.com. The rest of the instances are either terminated or running but not selected.

18. The result of dynamic application.

The screenshot shows a web application titled "FLIGHT PRICE" with a sunset background image of an airplane on a runway. The form includes fields for "Which Airline you want to travel?", "Source", "Destination", "Total Stops" (set to "Non-Stop"), and "Month". The "Which Airline" field has a dropdown menu open with the option "select you are airline". The browser status bar shows the URL 100.26.195.53:3000.

19. Here we can observe that the resources like VPC, Subnets, Route tables, Internet Gateway and created by using the playbook script.

❖ VPC.

The screenshot shows the AWS VPC dashboard with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#vpcs>. The dashboard lists two VPCs:

Name	VPC ID	State	Block Public Access	IPv4 CIDR	IPv6 CIDR
vinay-vpc	vpc-0fe790c7ea0e9238c	Available	Off	10.0.0.0/16	-
-	vpc-085ec71d994a9a327	Available	Off	172.31.0.0/16	-

The details for the first VPC (vinay-vpc) are shown in the expanded view:

VPC ID	State	Block Public Access	DNS hostnames
vpc-0fe790c7ea0e9238c	Available	Off	Enabled
DNS resolution	Tenancy	DHCP option set	Main route table
Enabled	Default	host-06-ef0d0c09efaf01-0	rtb-09-196674f270445

❖ Subnets.

The screenshot shows the AWS Subnets dashboard with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#subnets>. The dashboard lists subnets across three VPCs:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0350327d6f7894f1b	Available	vpc-085ec71d994a9a327	Off	172.31.80.0/16
vinay-pvt	subnet-0f01218495844f0aa	Available	vpc-0fe790c7ea0e9238c   vinay-vpc	Off	10.0.16.0/20
-	subnet-0ecbf1f3fab4a01c7	Available	vpc-085ec71d994a9a327	Off	172.31.32.0/20
-	subnet-09cbbd0b92842ce23	Available	vpc-085ec71d994a9a327	Off	172.31.0.0/20
-	subnet-0fa019b11702ec191	Available	vpc-085ec71d994a9a327	Off	172.31.16.0/20
-	subnet-040b9c0e422606bfa	Available	vpc-085ec71d994a9a327	Off	172.31.48.0/20
-	subnet-0f9874237f473da88	Available	vpc-085ec71d994a9a327	Off	172.31.16.0/20
vinay-pub	subnet-0f3d0662e238a4392	Available	vpc-0fe790c7ea0e9238c   vinay-vpc	Off	10.0.0.0/16

The details for the subnet `subnet-0f01218495844f0aa` under the `vinay-pvt` VPC are shown in the expanded view:

Subnet ID	State	VPC	Block Public...	IPv4 CIDR
subnet-0f01218495844f0aa	Available	vpc-0fe790c7ea0e9238c   vinay-vpc	Off	10.0.16.0/20

## ❖ Route tables.

The screenshot shows the AWS VPC Route Tables page. The left sidebar is the VPC dashboard, with the 'Route tables' section expanded. The main area displays a table of route tables:

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC
-	rtb-0b8a18663d3270d4a	-	-	Yes	vpc-0fe790c7ea0e9238c
-	rtb-08d790e48086a2fb	-	-	Yes	vpc-085ec71d994a9a327
<input checked="" type="checkbox"/> vinay-pvt-rt	rtb-0ebf71641347b6695	subnet-0f0121849584f...	-	No	vpc-0fe790c7ea0e9238c
<input checked="" type="checkbox"/> vinay-pub-rt	rtb-0b74c465fb027c3d8	subnet-0f3d0662e238a4...	-	No	vpc-0fe790c7ea0e9238c

Below the table, it says "Route tables: rtb-0ebf71641347b6695, rtb-0b74c465fb027c3d8".

## ❖ Internet Gateway.

The screenshot shows the AWS VPC Internet Gateways page. The left sidebar is the VPC dashboard, with the 'Internet gateways' section expanded. The main area displays a table of internet gateways:

Name	Internet gateway ID	State	VPC ID	Owner
<input checked="" type="checkbox"/> Vinay-igw	igw-0aee3de6a5f4a7d59	Attached	vpc-0fe790c7ea0e9238c   vinay-vpc	396913732678
-	igw-0fe9e5fd24c77548	Attached	vpc-085ec71d994a9a327	396913732678

Below the table, it says "igw-0aee3de6a5f4a7d59 / Vinay-igw".

The "Details" tab is selected, showing the following information:

Internet gateway ID igw-0aee3de6a5f4a7d59	State Attached	VPC ID vpc-0fe790c7ea0e9238c   vinay-vpc	Owner 396913732678
--	-------------------	---	-----------------------

## ❖ Security group.

20. And again write the playbook script for the deleting of the created resources by using playbook.

21. Create a file “delete.yaml”

22. And execute the command “ansible-playbook delete.yaml”

```
ec2-user@ip-172-31-88-89:~$ ansible-playbook deletetask.yaml

PLAY [Delete AWS resources] ****
TASK [Terminate EC2 instance 1] ****
changed: [ansible]

TASK [Delete the security group] ****
changed: [ansible]

TASK [Disassociate and Delete Public Route Table] ****
changed: [ansible]

TASK [Disassociate and Delete Private Route Table] ****
changed: [ansible]

TASK [Delete Public Subnet] ****
changed: [ansible]

TASK [Delete Private Subnet] ****
changed: [ansible]

TASK [Delete Internet Gateway] ****
changed: [ansible]

TASK [Delete VPC] ****
changed: [ansible]

PLAY RECAP ****
ansible      : ok=8    changed=8    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-88-89 ~]$
```

22.we can observe that the resources are deleted.

The screenshot shows two separate views of the AWS VPC console. The top view displays the 'Your VPCs' table with one entry:

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
-	vpc-085ec71d994a9a327	Available	Off	172.31.0.0/16	-

The bottom view displays the 'Subnets' table with six entries:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0350327d6f7894f1b	Available	vpc-085ec71d994a9a327	Off	172.31.80.0/24
-	subnet-0eefbf1f3fab4a01c7	Available	vpc-085ec71d994a9a327	Off	172.31.32.0/24
-	subnet-09cbbd0b92842ce23	Available	vpc-085ec71d994a9a327	Off	172.31.0.0/24
-	subnet-0fad19b11702ec191	Available	vpc-085ec71d994a9a327	Off	172.31.64.0/24
-	subnet-040b9c0e422606bf	Available	vpc-085ec71d994a9a327	Off	172.31.48.0/24
-	subnet-0f9874237f473da88	Available	vpc-085ec71d994a9a327	Off	172.31.16.0/24

A small screenshot of a terminal window titled 'vpc creation ansible.txt - Notepad...' is visible at the bottom right, showing some Ansible configuration code.

The screenshot shows the AWS CloudWatch Metrics interface. A metric named "AWS Lambda Function Invocations" is displayed with a value of 1. The chart shows a single data series over time from 2024-05-12T00:00:00Z to 2024-05-12T16:00:00Z. The Y-axis ranges from 0 to 1000. The X-axis shows hours from 00:00 to 16:00.

23.this the playbook script for the deletion.

```

- name: Delete AWS resources
  hosts: localhost
  gather_facts: no
  collections:
    - amazon.aws
  vars:
    aws_access_key: "AKIAVY2PG5R005KUAWCJ"
    aws_secret_key: "fH9F8UPif2Iu9zdx0DQczJXMIGWFN9y+nikSE5Gk"
  tasks:
    - name: Terminate EC2 instance 1
      ec2_instance:
        instance_ids:
          - i-0977fda48e9209a35
          - i-0becf9a78360ad14c # Replace with your instance ID
        region: us-east-1
        state: absent
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"

    - name: Delete the security group
      ec2_group:
        group_id: "sg-050368f253cd9caca"
        region: us-east-1
        state: absent
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"

    - name: Disassociate and Delete Public Route Table
      ec2_vpc_route_table:
        region: us-east-1
        vpc_id: "vpc-0fe790c7ea0e9238c"
        state: absent
        tags:
          Name: "vinay-pub-rt"
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"

    - name: Disassociate and Delete Private Route Table
      ec2_vpc_route_table:
        region: us-east-1
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"

"deletetask.yaml" 82L, 2410B
1,28 Top

```

```
ec2-user@ip-172-31-88-89: ~ % Windows PowerShell x Windows PowerShell x + v
- name: Disassociate and Delete Public Route Table
  ec2_vpc_route_table:
    region: us-east-1
    vpc_id: "vpc-0fe790c7ea0e9238c"
    state: absent
    tags:
      Name: "vinay-pub-rt"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Disassociate and Delete Private Route Table
  ec2_vpc_route_table:
    region: us-east-1
    vpc_id: "vpc-0fe790c7ea0e9238c"
    state: absent
    tags:
      Name: "vinay-pvt-rt"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Public Subnet
  ec2_vpc_subnet:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    cidr: 10.0.0.0/20
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Private Subnet
  ec2_vpc_subnet:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    cidr: 10.0.16.0/20
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

  61,25 61%

```

```
tags:
  Name: "vinay-pvt-rt"
  aws_access_key: "{{ aws_access_key }}"
  aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Public Subnet
  ec2_vpc_subnet:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    cidr: 10.0.0.0/20
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Private Subnet
  ec2_vpc_subnet:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    cidr: 10.0.16.0/20
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "vpc-0fe790c7ea0e9238c"
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

- name: Delete VPC
  ec2_vpc_net:
    cidr_block: 10.0.0.0/16
    name: vinay-vpc
    region: us-east-1
    state: absent
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

  82,0-1 Bot

```

## Method-2 (Inline Variables)

- ❖ Creating the aws resources and hosting the static and dynamic application by using the Inline variable for playbook script.

- Launch an instance using amazon-linux.

- Connect it to the terminal and update it.
- Then install ansible.

```
ec2-user@ip-172-31-29-229: ~ + ~
  _###_ Amazon Linux 2
  _\###\ AL2 End of Life is 2025-06-30.
  \###_
  \#_ .-+>
  _/_ A newer version of Amazon Linux is available!
  _/_ / Amazon Linux 2023, GA and supported until 2028-03-15.
  _/_ / https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-29-229 ~]$ ansible: command not found
[ec2-user@ip-172-31-29-229 ~]$ sudo amazon-linux-extras install ansible -y
Topic ansible is not found.
[ec2-user@ip-172-31-29-229 ~]$ sudo amazon-linux-extras install ansible2 -y
Installing ansible
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-ansible2 amzn2extra-docker amzn2extra-kernel-5.10
9 metadata files removed
0 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-ansible2
amzn2extra-docker
amzn2extra-kernel-5.10
(1/9): amzn2-core/2/x86_64/group.gz
(2/9): amzn2-core/2/x86_64/updateinfo
(3/9): amzn2extra-docker/2/x86_64/primary_db
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo
(5/9): amzn2extra-ansible2/2/x86_64/updateinfo
(6/9): amzn2extra-ansible2/2/x86_64/primary_db
(7/9): amzn2extra-docker/2/x86_64/updateinfo
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db
(9/9): amzn2-core/2/x86_64/primary_db
Resolving Dependencies
--> Running transaction check
--> Package ansible.noarch 0:2.9.23-1.amzn2 will be installed
--> Processing Dependency: python-crypto for package: ansible-2.9.23-1.amzn2.noarch
--> Processing Dependency: python-httplib2 for package: ansible-2.9.23-1.amzn2.noarch
--> Processing Dependency: python-keyczar for package: ansible-2.9.23-1.amzn2.noarch
| 3.6 kB 00:00:00
| 2.9 kB 00:00:00
| 2.9 kB 00:00:00
| 3.0 kB 00:00:00
| 2.7 kB 00:00:00
| 1.0 MB 00:00:00
| 116 kB 00:00:00
| 93 kB 00:00:00
| 4.7 kB 00:00:00
| 37 kB 00:00:00
| 20 kB 00:00:00
| 32 MB 00:00:00
| 72 MB 00:00:01
[ec2-user@ip-172-31-29-229 ~]$
```

- Edit the configure file.

```

1 # config file for ansible -- https://ansible.com/
2 # =====
3
4 # nearly all parameters can be overridden in ansible-playbook
5 # or with command line flags, ansible will read ANSIBLE_CONFIG,
6 # ansible.cfg in the current working directory, .ansible.cfg in
7 # the home directory or /etc/ansible/ansible.cfg, whichever it
8 # finds first
9
10 [defaults]
11
12 # some basic default values...
13
14 inventory      = /etc/ansible/hosts
15 #library       = /usr/share/my_modules/
16 #module_utils  = /usr/share/my_module_utils/
17 #remote_tmp    = ~/.ansible/tmp
18 #local_tmp     = ~/.ansible/tmp
19 #plugin_filters_cfg = /etc/ansible/plugin_filters.yml
20 #forks         = 5
21 #poll_interval = 15
22 #sudo_user    = root
23 #ask_sudo_pass = True
24 #ask_pass     = True
25 #transport   = smart
26 #remote_port = 22
27 #module_lang  = C
28 #module_set_locale = False
29
30 # plays will gather facts by default, which contain information about
31 # the remote system.
32 #
33 # smart - gather by default, but don't regather if already gathered
34 # implicit - gather by default, turn off with gather_facts: False
35 # explicit - do not gather by default, must say gather_facts: True
36 #gathering = implicit
37
38 # This only affects the gathering done by a play's gather_facts directive,
39 # by default gathering retrieves all facts subsets
40 # all - gather all subsets
41 # network - gather min and network facts
42 # hardware - gather hardware facts (longest facts to retrieve)
43 # virtual - gather min and virtual facts
44 # facter - import facts from facter
45 # ohai - import facts from ohai
46 # You can combine them using comma (ex: network,virtual)
47 # You can negate them using ! (ex: !hardware,!facter,!ohai)
48 # A minimal set of facts is always gathered.
49 #gather_subset = all
50
51 # some hardware related facts are collected
52 # with a maximum timeout of 10 seconds. This
53 # option lets you increase or decrease that
54 # timeout to something more suitable for the
55 # environment.
56 # gather_timeout = 10
57
58 # Ansible facts are available inside the ansible.facts.* dictionary
59 # namespace. This setting maintains the behaviour which was the default prior
60 # to 2.5, duplicating these variables into the main namespace, each with a
61 # prefix of 'ansible_'.
62 # This variable is set to True by default for backwards compatibility. It
63 # will be changed to a default of 'False' in a future release.
64 # ansible_facts
65 # inject_facts_as_vars = True
66
67 # additional paths to search for roles in, colon separated
68 #roles_path     = /etc/ansible/roles
69
70 # uncomment this to disable SSH key host checking
71 host_key_checking = False
72
73 # change the default callback, you can only have one 'stdout' type enabled at a time.
74 #stdout_callback = skippy
75
76
77 ## Ansible ships with some plugins that require whitelisting,
-- INSERT --

```

1,1 Top

28°C Haze ENG IN 14:18 05-12-2024

```

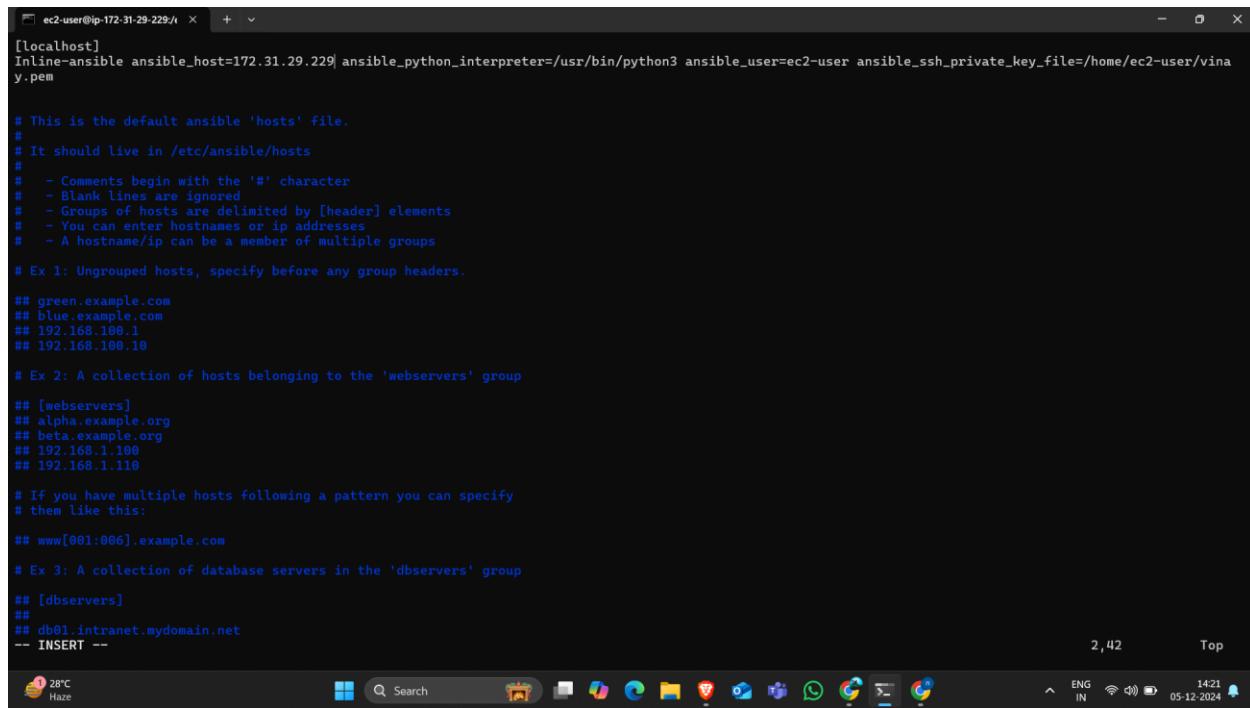
38 # This only affects the gathering done by a play's gather_facts directive,
39 # by default gathering retrieves all facts subsets
40 # all - gather all subsets
41 # network - gather min and network facts
42 # hardware - gather hardware facts (longest facts to retrieve)
43 # virtual - gather min and virtual facts
44 # facter - import facts from facter
45 # ohai - import facts from ohai
46 # You can combine them using comma (ex: network,virtual)
47 # You can negate them using ! (ex: !hardware,!facter,!ohai)
48 # A minimal set of facts is always gathered.
49 #gather_subset = all
50
51 # some hardware related facts are collected
52 # with a maximum timeout of 10 seconds. This
53 # option lets you increase or decrease that
54 # timeout to something more suitable for the
55 # environment.
56 # gather_timeout = 10
57
58 # Ansible facts are available inside the ansible.facts.* dictionary
59 # namespace. This setting maintains the behaviour which was the default prior
60 # to 2.5, duplicating these variables into the main namespace, each with a
61 # prefix of 'ansible_'.
62 # This variable is set to True by default for backwards compatibility. It
63 # will be changed to a default of 'False' in a future release.
64 # ansible_facts
65 # inject_facts_as_vars = True
66
67 # additional paths to search for roles in, colon separated
68 #roles_path     = /etc/ansible/roles
69
70 # uncomment this to disable SSH key host checking
71 host_key_checking = False
72
73 # change the default callback, you can only have one 'stdout' type enabled at a time.
74 #stdout_callback = skippy
75
76
77 ## Ansible ships with some plugins that require whitelisting,
-- INSERT --

```

71,1 8%

28°C Haze ENG IN 14:18 05-12-2024

- Edit the hosts file.



```
[localhost]
Inline-ansible ansible_host=172.31.29.229 ansible_python_interpreter=/usr/bin/python3 ansible_user=ec2-user ansible_ssh_private_key_file=/home/ec2-user/vinay.pem

# This is the default ansible 'hosts' file.
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.18

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

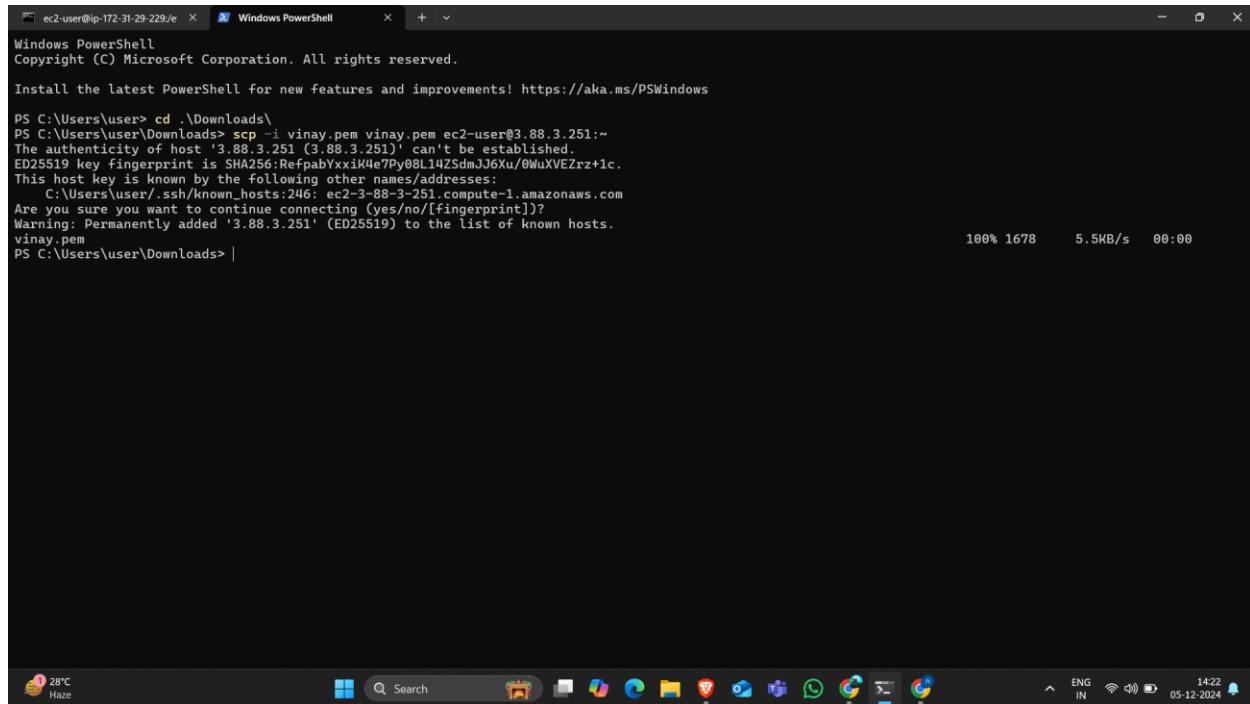
## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
## db01.intranet.mydomain.net
-- INSERT --
```

The screenshot shows a terminal window titled "ec2-user@ip-172-31-29-229:/". The window displays the content of an Ansible hosts file. It includes examples for ungrouped hosts, hosts grouped under 'webservers', and hosts grouped under 'dbservers'. The terminal also shows system status icons at the bottom, including battery level (28°C Haze), network connectivity, and system date/time (14:21 05-12-2024).

- Copy the pem file and give execute permissions to the pem file.



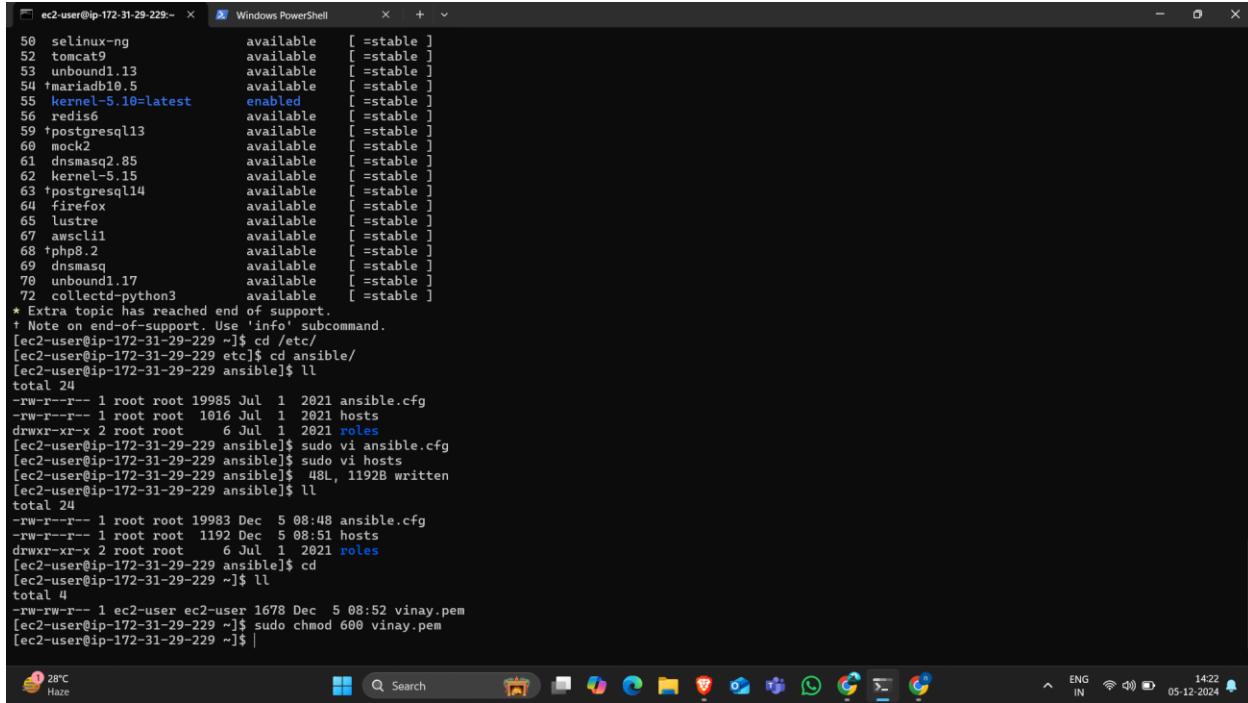
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\user> cd ..\Downloads
PS C:\Users\user\Downloads> scp -i vinay.pem vinay.pem ec2-user@3.88.3.251:~
The authenticity of host '3.88.3.251 (3.88.3.251)' can't be established.
ED25519 key fingerprint is SHA256:RefpabYxx1K4e7Py88L14ZSdmJJ6Xu/0WuXVErz+1c.
This host key is known by the following other names/addresses:
  C:\Users\user/.ssh/known_hosts:246: ec2-3-88-3-251.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added '3.88.3.251' (ED25519) to the list of known hosts.
vinay.pem
PS C:\Users\user\Downloads> |
```

The screenshot shows a terminal window titled "Windows PowerShell" running on a Windows operating system. It displays the output of a "scp" command used to copy a file named "vinay.pem" from the local machine to an EC2 instance. The command includes the "-i" option to specify the private key file. The terminal also shows system status icons at the bottom, including battery level (28°C Haze), network connectivity, and system date/time (14:22 05-12-2024).

- Give the permission to the pem file.



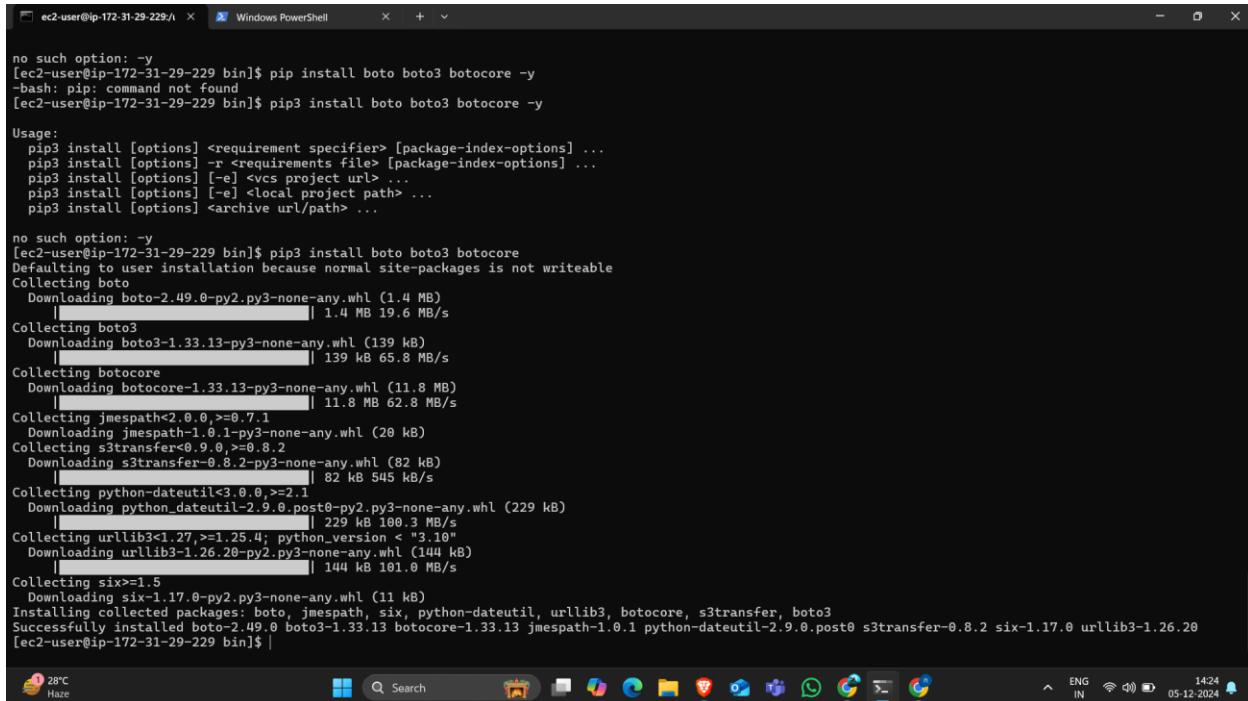
```

ec2-user@ip-172-31-29-229:~ % Windows PowerShell x + v
50 selinux-ng available [ =stable ]
52 tomcat9 available [ =stable ]
53 unbound1.13 available [ =stable ]
54 tmariadb10.5 available [ =stable ]
55 kernel-5.10=latest enabled [ =stable ]
56 redis6 available [ =stable ]
59 +postgresql13 available [ =stable ]
60 mock2 available [ =stable ]
61 dnsmasq2.85 available [ =stable ]
62 kernel-5.15 available [ =stable ]
63 +postgresql14 available [ =stable ]
64 firefox available [ =stable ]
65 lustre available [ =stable ]
67 awscli1 available [ =stable ]
68 +php8.2 available [ =stable ]
69 dnsmasq available [ =stable ]
70 unbound1.17 available [ =stable ]
72 collectd-python3 available [ =stable ]
* Extra topic has reached end of support.
+ Note on end-of-support. Use 'info' subcommand.
[ec2-user@ip-172-31-29-229 ~]$ cd /etc/
[ec2-user@ip-172-31-29-229 etc]$ cd ansible/
[ec2-user@ip-172-31-29-229 ansible]$ ll
total 24
-rw-r--r-- 1 root root 19985 Jul 1 2021 ansible.cfg
-rw-r--r-- 1 root root 1016 Jul 1 2021 hosts
drwxr-xr-x 2 root root 6 Jul 1 2021 roles
[ec2-user@ip-172-31-29-229 ansible]$ sudo vi ansible.cfg
[ec2-user@ip-172-31-29-229 ansible]$ sudo vi hosts
[ec2-user@ip-172-31-29-229 ansible]$ 48L, 1192B written
[ec2-user@ip-172-31-29-229 ansible]$ ll
total 4
-rw-r--r-- 1 root root 19983 Dec 5 08:48 ansible.cfg
-rw-r--r-- 1 root root 1192 Dec 5 08:51 hosts
drwxr-xr-x 2 root root 6 Jul 1 2021 roles
[ec2-user@ip-172-31-29-229 ansible]$ cd
[ec2-user@ip-172-31-29-229 ~]$ ll
total 4
-rw-r--r-- 1 ec2-user ec2-user 1678 Dec 5 08:52 vinay.pem
[ec2-user@ip-172-31-29-229 ~]$ sudo chmod 600 vinay.pem
[ec2-user@ip-172-31-29-229 ~]$ |

```

The screenshot shows a Windows PowerShell window with the title bar "ec2-user@ip-172-31-29-229:~" and "Windows PowerShell". The command history shows the user navigating through Ansible's configuration files and hosts, then using the `vi` editor to edit `ansible.cfg`. After saving changes, they run `ll` to list the contents of the directory. They then use `sudo` to change the permissions of the `vinay.pem` file to 600. The taskbar at the bottom includes icons for File Explorer, Task View, and various system status indicators like battery level and signal strength. The system tray shows the date and time as "05-12-2024 14:22".

- Install the boto boto3 botocore.



```

no such option: -y
[ec2-user@ip-172-31-29-229 bin]$ pip install boto boto3 botocore -y
bash: pip: command not found
[ec2-user@ip-172-31-29-229 bin]$ pip3 install boto boto3 botocore -y

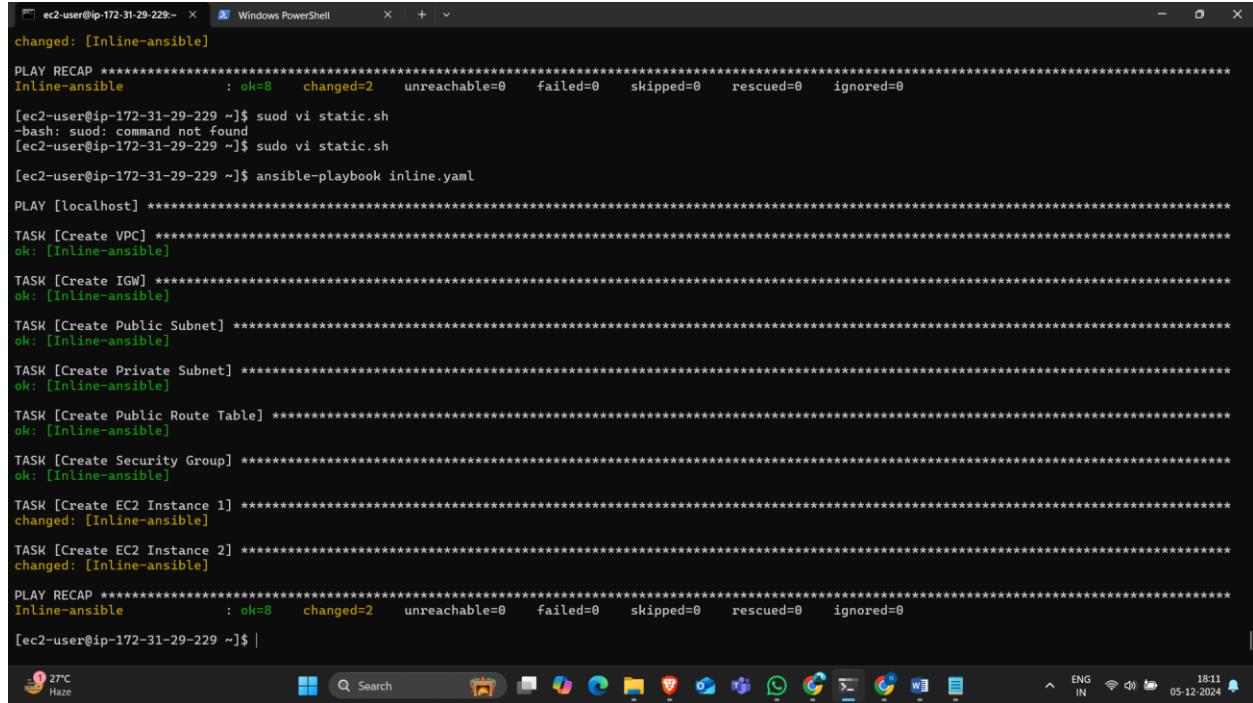
Usage:
  pip3 install [options] <requirement specifier> [package-index-options] ...
  pip3 install [options] -r <requirements file> [package-index-options] ...
  pip3 install [options] [-e] <vcs project url> ...
  pip3 install [options] [-e] <local project path> ...
  pip3 install [options] <archive url/path> ...

no such option: -y
[ec2-user@ip-172-31-29-229 bin]$ pip3 install boto boto3 botocore
Defaulting to user installation because normal site-packages is not writeable
Collecting boto
  Downloading boto-2.49.0-py2.py3-none-any.whl (1.4 MB)
    |██████████| 1.4 MB 19.6 MB/s
Collecting boto3
  Downloading boto3-1.33.13-py3-none-any.whl (139 kB)
    |██████████| 139 kB 65.8 MB/s
Collecting botocore
  Downloading botocore-1.33.13-py3-none-any.whl (11.8 MB)
    |██████████| 11.8 MB 62.8 MB/s
Collecting jmespath<2.0.0,>=0.7.1
  Downloading jmespath-1.0.1-py3-none-any.whl (20 kB)
Collecting s3transfer<0.9.0,>=0.8.2
  Downloading s3transfer-0.8.2-py3-none-any.whl (82 kB)
    |██████████| 82 kB 545 kB/s
Collecting python-dateutil<3.0.0,>=2.1
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
    |██████████| 229 kB 100.3 MB/s
Collecting urllib3<1.27,>=1.25.4; python_version < "3.10"
  Downloading urllib3-1.26.20-py2.py3-none-any.whl (144 kB)
    |██████████| 144 kB 101.0 MB/s
Collecting six>=1.5
  Downloading six-1.17.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: boto, jmespath, six, python-dateutil, urllib3, botocore, s3transfer, boto3
Successfully installed boto-2.49.0 boto3-1.33.13 botocore-1.33.13 jmespath-1.0.1 python-dateutil-2.9.0.post0 s3transfer-0.8.2 six-1.17.0 urllib3-1.26.20
[ec2-user@ip-172-31-29-229 bin]$ |

```

The screenshot shows a Windows PowerShell window with the title bar "ec2-user@ip-172-31-29-229:~" and "Windows PowerShell". The command history shows the user attempting to install `boto`, `boto3`, and `botocore` using `pip` and then switching to `pip3` to achieve the same result. The process fails initially due to a missing `pip` command, but succeeds when using `pip3`. The output shows the download and installation of several dependencies, including `jmespath`, `s3transfer`, and `python-dateutil`. The taskbar at the bottom includes icons for File Explorer, Task View, and various system status indicators like battery level and signal strength. The system tray shows the date and time as "05-12-2024 14:24".

- Create the playbook file and run the the file “ansible-playbook.yaml”



```

ec2-user@ip-172-31-29-229:~ % Windows PowerShell
changed: [Inline-ansible]

PLAY RECAP ****
Inline-ansible : ok=8    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-29-229 ~]$ suod vi static.sh
-bash: suod: command not found
[ec2-user@ip-172-31-29-229 ~]$ sudo vi static.sh

[ec2-user@ip-172-31-29-229 ~]$ ansible-playbook inline.yaml

PLAY [localhost] ****
TASK [Create VPC] ****
ok: [Inline-ansible]

TASK [Create IGW] ****
ok: [Inline-ansible]

TASK [Create Public Subnet] ****
ok: [Inline-ansible]

TASK [Create Private Subnet] ****
ok: [Inline-ansible]

TASK [Create Public Route Table] ****
ok: [Inline-ansible]

TASK [Create Security Group] ****
ok: [Inline-ansible]

TASK [Create EC2 Instance 1] ****
changed: [Inline-ansible]

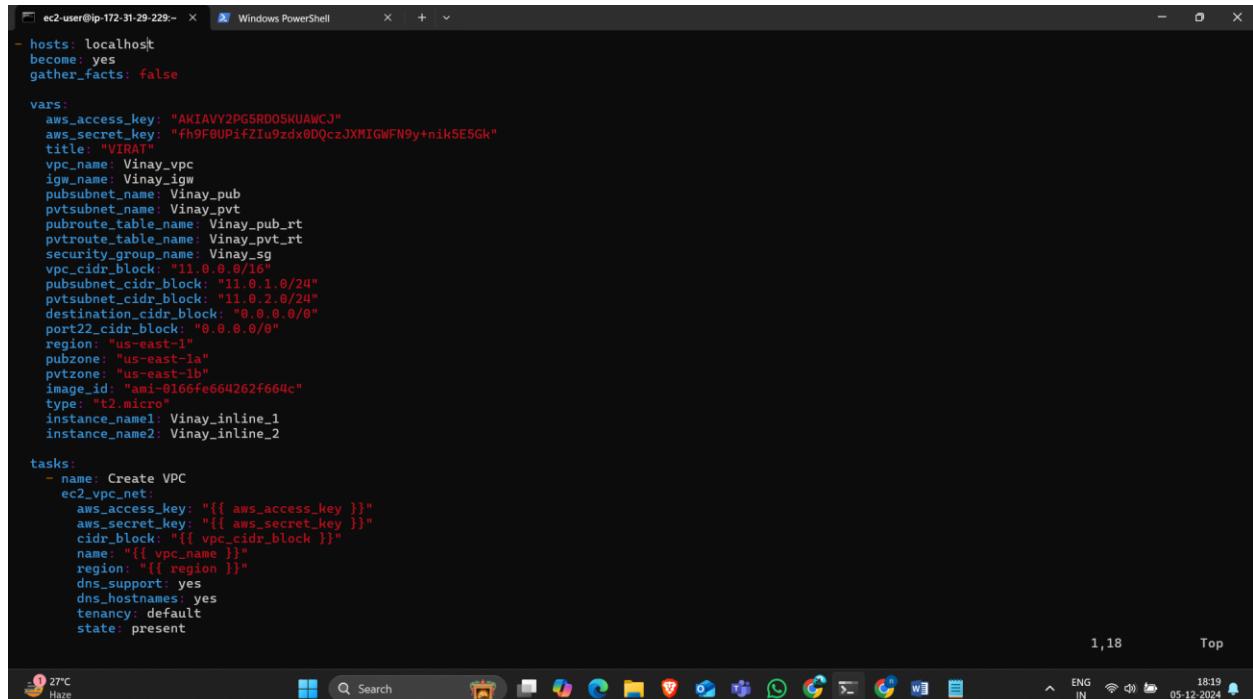
TASK [Create EC2 Instance 2] ****
changed: [Inline-ansible]

PLAY RECAP ****
Inline-ansible : ok=8    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-29-229 ~] |

```

- Here is the script for the Inline variable and resources.



```

hosts: localhost
become: yes
gather_facts: false

vars:
  aws_access_key: "AKIAVY2PG5RD05UAWCJ"
  aws_secret_key: "fh9F0UPifZiu9zdx0DQczJXMIGWFN9y+nik5E5Gk"
  title: "VIRAT"
  vpc_name: Vinay_vpc
  igw_name: Vinay_igw
  pubsubnet_name: Vinay_pub
  pvtsubnet_name: Vinay_pvt
  pubroute_table_name: Vinay_pub_rt
  pvtroute_table_name: Vinay_pvt_rt
  security_group_name: Vinay_sg
  vpc_cidr_block: "11.0.0.0/16"
  pubsubnet_cidr_block: "11.0.1.0/24"
  pvtsubnet_cidr_block: "11.0.2.0/24"
  destination_cidr_block: "0.0.0.0/0"
  port22_cidr_block: "0.0.0.0/0"
  region: "us-east-1"
  pubzone: "us-east-1a"
  pvtzone: "us-east-1b"
  image_id: "ami-0166fe664262f664c"
  type: "t2.micro"
  instance_name1: Vinay_inline_1
  instance_name2: Vinay_inline_2

tasks:
  - name: Create VPC
    ec2_vpc.net:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      cidr_block: "{{ vpc_cidr_block }}"
      name: "{{ vpc_name }}"
      region: "{{ region }}"
      dns_support: yes
      dns_hostnames: yes
      tenancy: default
      state: present

```

```
ec2-user@ip-172-31-29-229:~ % Windows PowerShell + - x

tasks:
  - name: Create VPC
    ec2_vpc_net:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      cidr_block: "{{ vpc_cidr_block }}"
      name: "{{ vpc_name }}"
      region: "{{ region }}"
      dns_support: yes
      dns_hostnames: yes
      tenancy: default
      state: present
      register: vpc_result

  - name: Create IGW
    ec2_vpc_igw:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      vpc_id: "{{ vpc_result.vpc.id }}"
      region: "{{ region }}"
      state: present
      tags:
        Name: "{{ igw_name }}"
      register: igw_result

  - name: Create Public Subnet
    ec2_vpc_subnet:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      vpc_id: "{{ vpc_result.vpc.id }}"
      region: "{{ region }}"
      az: "{{ pubzone }}"
      state: present
      cidr: "{{ pubsubnet_cidr_block }}"
      map_public: yes
      resource_tags:
        Name: "{{ pubsubnet_name }}"
      register: pubsubnet_result

62,42 25%
```

```
register: pubsubnet_result

  - name: Create Private Subnet
    ec2_vpc_subnet:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      vpc_id: "{{ vpc_result.vpc.id }}"
      region: "{{ region }}"
      az: "{{ pvtzone }}"
      state: present
      cidr: "{{ pvtsubnet_cidr_block }}"
      map_public: no
      resource_tags:
        Name: "{{ pvtsubnet_name }}"
      register: pvtsubnet_result

  - name: Create Public Route Table
    ec2_vpc_route_table:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      vpc_id: "{{ vpc_result.vpc.id }}"
      region: "{{ region }}"
      state: present
      tags:
        Name: "{{ pubroute_table_name }}"
      subnets: [ "{{ pubsubnet_result.subnet.id }} " ]
      routes:
        - dest: "{{ destination_cidr_block }}"
          gateway_id: "{{ igw_result.gateway_id }}"
      register: public_route_table

  - name: Create Security Group
    ec2_group:
      aws_access_key: "{{ aws_access_key }}"
      aws_secret_key: "{{ aws_secret_key }}"
      vpc_id: "{{ vpc_result.vpc.id }}"
      region: "{{ region }}"
      state: present
      name: "{{ security_group_name }}"
      description: "Allow all traffic"
      register: security_group

100,45 69%
```

```
ec2-user@ip-172-31-29-229: ~ Windows PowerShell + x
- proto: -1
  cidr_ip: "[{ port22_cidr_block }]"
  rule_desc: Allow all traffic
register: security_group_results

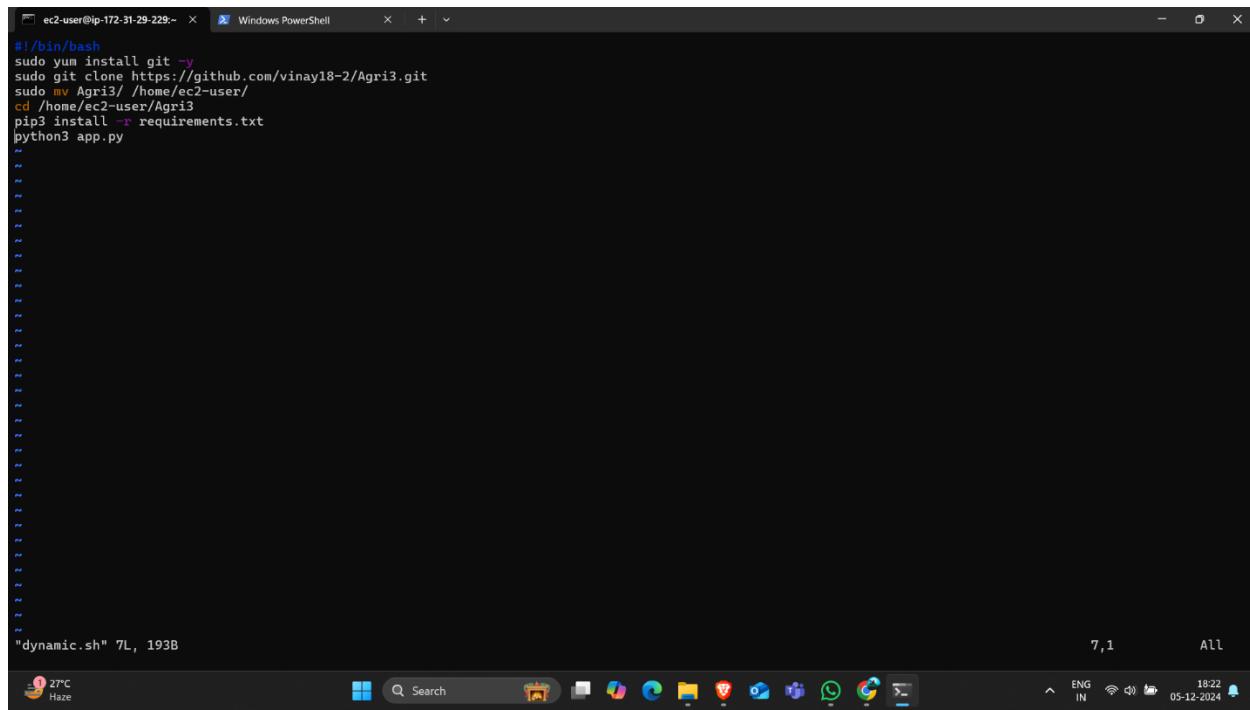
- name: Create EC2 Instance 1
ec2:
  image: "{{ image_id }}"
  instance_type: "{{ type }}"
  region: "{{ region }}"
  wait: yes
  count: 1
  state: present
  vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}" # Correct parameter
  assign_public_ip: yes
  group_id: "{{ security_group_results.group_id }}"
  aws_access_key: "{{ aws_access_key }}"
  aws_secret_key: "{{ aws_secret_key }}"
  user_data: "{{ lookup('file', 'static.sh') }}"
  instance_tags:
    Name: "{{ instance_name1 }}"

- name: Create EC2 Instance 2
ec2:
  image: "{{ image_id }}"
  instance_type: "{{ type }}"
  region: "{{ region }}"
  wait: yes
  count: 1
  state: present
  vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}" # Correct parameter
  assign_public_ip: yes # Private subnet
  group_id: "{{ security_group_results.group_id }}"
  aws_access_key: "{{ aws_access_key }}"
  aws_secret_key: "{{ aws_secret_key }}"
  user_data: "{{ lookup('file', 'dynam[ec].sh') }}"
  instance_tags:
    Name: "{{ instance_name2 }}"

145,45 ENG IN 18:20 05-12-2024
```

- Static.sh file.

- Dynamic.sh file.

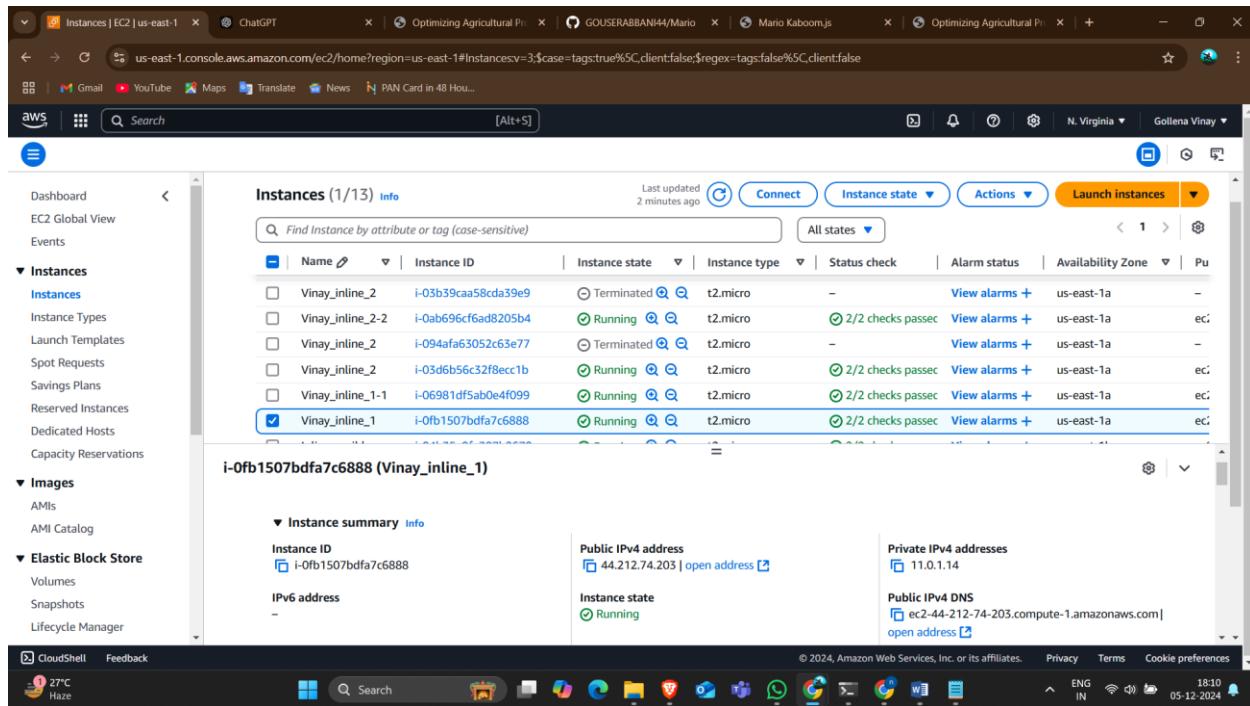


```
#!/bin/bash
sudo yum install git -y
sudo git clone https://github.com/vinayl8-2/Agri3.git
sudo mv Agri3/ /home/ec2-user/
cd /home/ec2-user/Agri3
pip3 install -r requirements.txt
python3 app.py

"dynamic.sh" 7L, 193B
```

- ❖ Here the created resources results.

### 1. Instance-1



**Instances (1/13) Info**

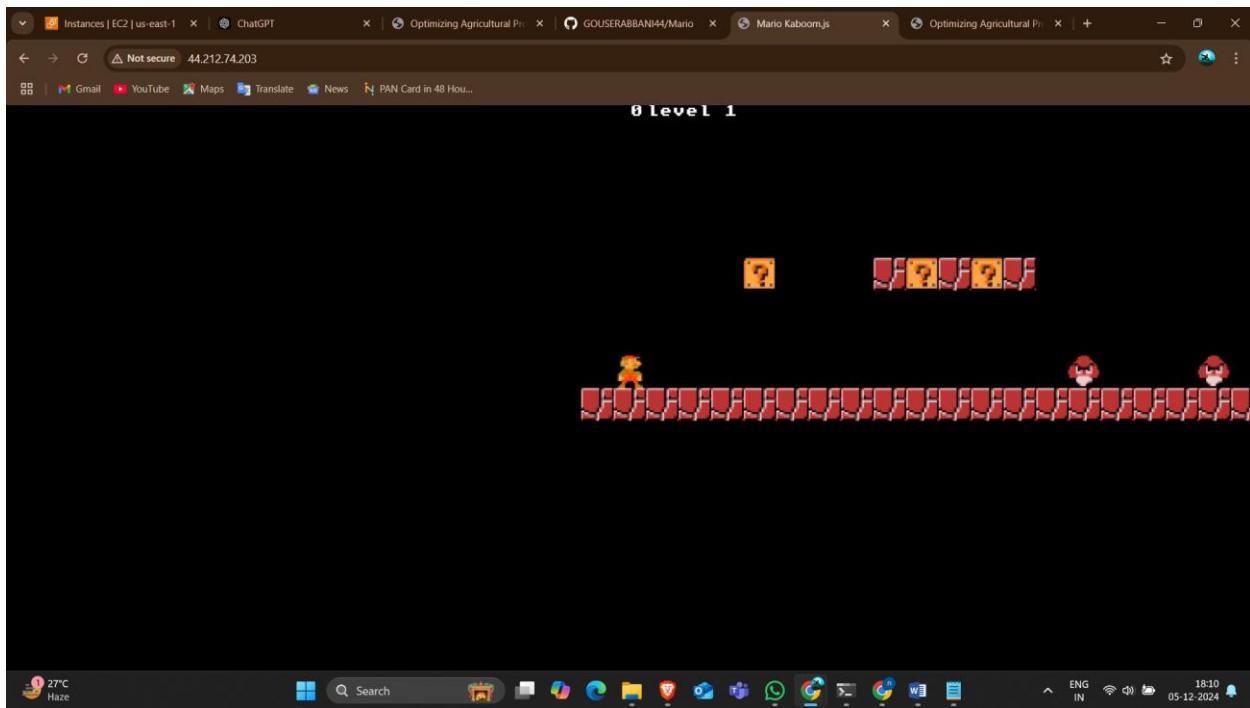
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
Vinay_inline_2	i-03b59caa58cda39e9	Terminated	t2.micro	-	View alarms +	us-east-1a	-
Vinay_inline_2-2	i-0ab696cf6ad8205b4	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
Vinay_inline_2	i-094fa63052c65e77	Terminated	t2.micro	-	View alarms +	us-east-1a	-
Vinay_inline_2	i-03d6b56c32f8ecc1b	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
Vinay_inline_1-1	i-06981df5ab0e4f099	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
<b>Vinay_inline_1</b>	<b>i-0fb1507bdfa7c6888</b>	<b>Running</b>	<b>t2.micro</b>	<b>2/2 checks passed</b>	<b>View alarms +</b>	<b>us-east-1a</b>	<b>ec2</b>

**i-0fb1507bdfa7c6888 (Vinay\_inline\_1)**

**Instance summary**

Instance ID	i-0fb1507bdfa7c6888	Public IPv4 address	44.212.74.203   open address
IPv6 address	-	Instance state	Running
		Private IPv4 addresses	11.0.1.14
		Public IPv4 DNS	ec2-44-212-74-203.compute-1.amazonaws.com   open address

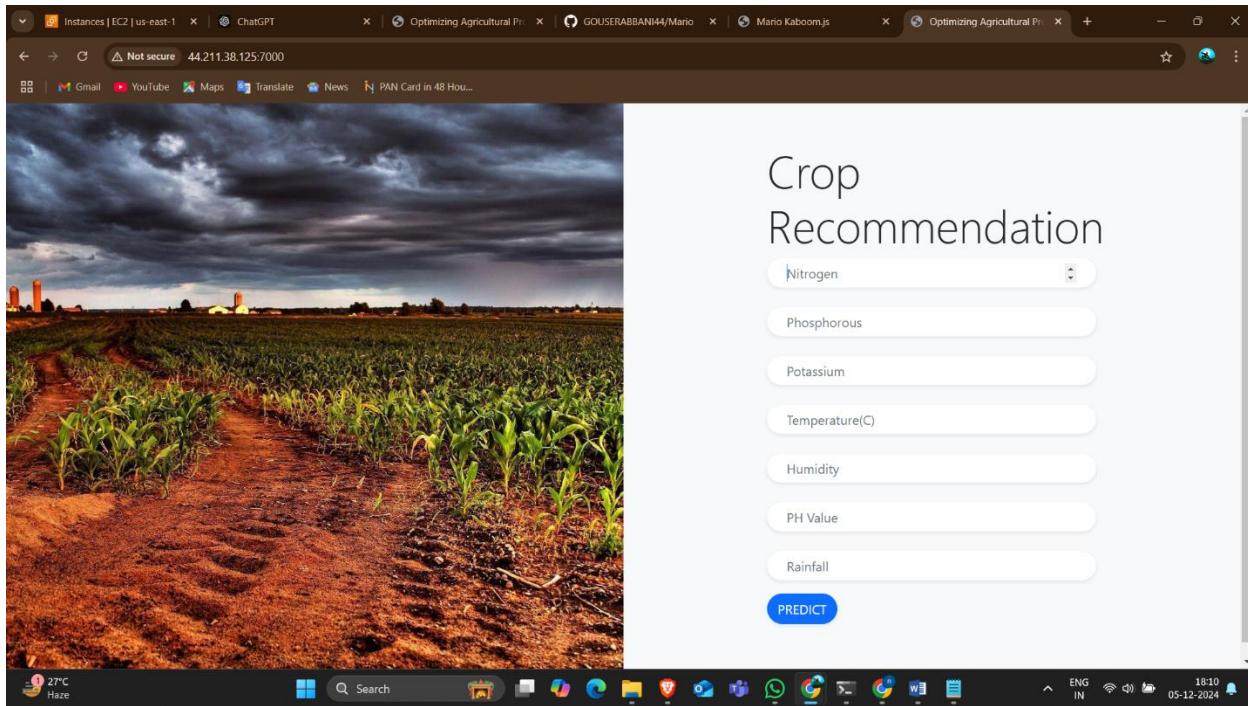
## 2. Result os static website in instance-1.



## 3. Result of instance-2.

A screenshot of the AWS Management Console, specifically the EC2 Instances page. The left sidebar shows navigation options like Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. The main area displays a table titled "Instances (1/13) Info" with 13 rows. One row is selected, showing an instance named "Vinay\_inline\_2" with the ID i-03d6b56c32f8ecc1b, which is currently "Running". Other columns include Instance ID, Instance state, Instance type (t2.micro), Status check, Alarm status, Availability Zone (us-east-1a or us-east-1b), and a Launch instances button. Below the table, a detailed view for the selected instance is shown, including sections for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The status bar at the bottom indicates the user is in N. Virginia and the date is 05-12-2024.

#### 4. Hosted dynamic application.



- Then write the deletion playbook file to delete the created resources.
- Execute that file.

```
[ec2-user@ip-172-31-29-229 ~]$ sudo vi delete1.yaml
[ec2-user@ip-172-31-29-229 ~]$ ansible-playbook delete1.yaml

PLAY [localhost] ****
TASK [Gathering Facts] ****
ok: [Inline-ansible]

TASK [Terminate EC2 Instance] ****
changed: [Inline-ansible]

TASK [Delete Security Group] ****
changed: [Inline-ansible]

TASK [Delete Public Subnet] ****
changed: [Inline-ansible]

TASK [Delete Private Subnet] ****
changed: [Inline-ansible]

TASK [Delete Public Route Table] ****
changed: [Inline-ansible]

TASK [Delete Private Route Table] ****
ok: [Inline-ansible]

TASK [Detach and Delete Internet Gateway] ****
changed: [Inline-ansible]

TASK [Delete VPC by name] ****
changed: [Inline-ansible]

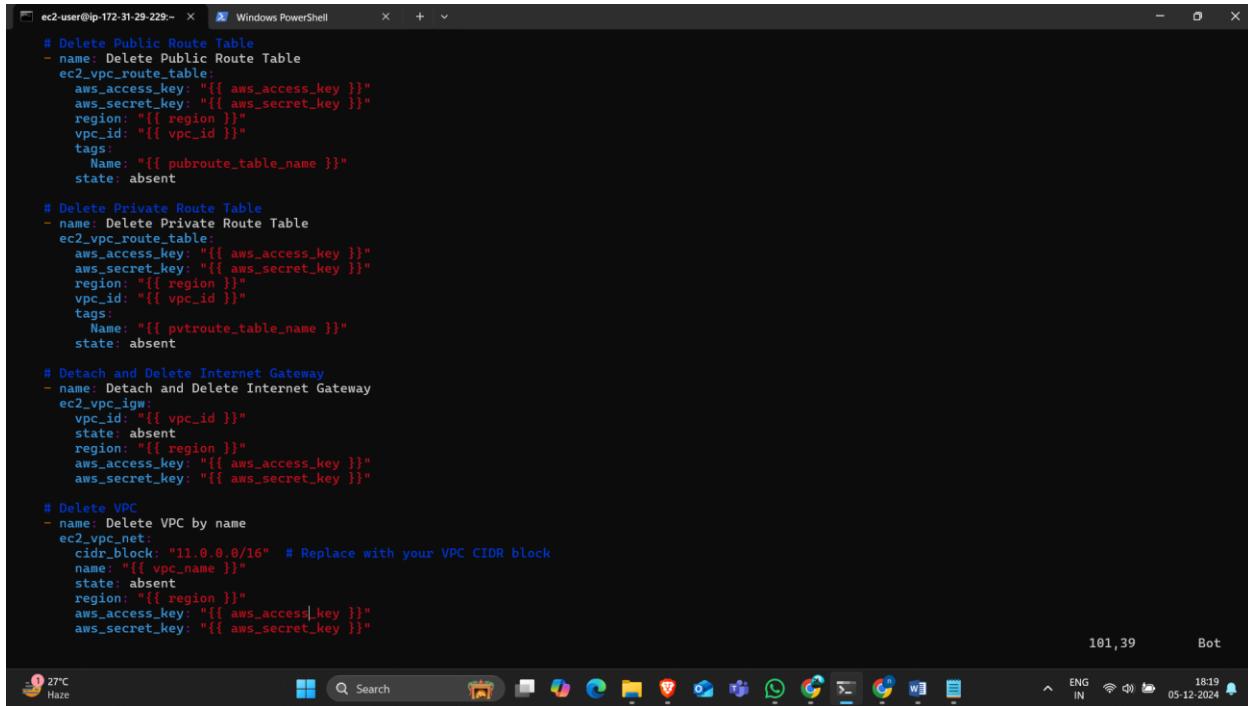
PLAY RECAP ****
Inline-ansible      : ok=9   changed=7    unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

[ec2-user@ip-172-31-29-229 ~]$ sudo vi delete1.yaml
[ec2-user@ip-172-31-29-229 ~]$ sudo vi inline.yaml
[ec2-user@ip-172-31-29-229 ~]$ cat inline.yaml
- hosts: localhost
  become: yes
```

- Script for deletion.

```
ec2-user@ip-172-31-29-229:~ % Windows PowerShell
-
hosts: localhost
  vars:
    aws_access_key: "AKIAVY2PG5R005MUAWCJ"
    aws_secret_key: "f9f9f8UP1fZIu9zdx6DQczjXMIGWFN9y+nik5E5Gk"
    region: "us-east-1"
    vpc_id: "vpc-0f394f168bc03e9ad"
    pubsubnet_cidr_block: "11.0.1.0/24"
    pvtsubnet_cidr_block: "11.0.2.0/24"
    instance_ids:
      - "i-0320ef9fad8ba9b20"
      - "i-025344642a9db007"
      - "i-0fb1507bdffa7c6888"
      - "i-06981df5ab0e4f099"
      - "i-03d6b56c32f8ecc1b"
      - "i-0ab696cfaad8205b4"
    # Replace with your instance IDs
    security_group_name: "Vinay_sg"
    security_group_id: "sg-08e988361b657b812" # Replace with your Security Group ID
    pubroute_table_name: "Vinay_pub_rt"
    pvtroute_table_name: "Vinay_pvt_rt"
    igw_name: "Vinay_igw"
    vpc_name: "Vinay_vpc"
  tasks:
    # Terminate EC2 Instances
    - name: Terminate EC2 Instance
      ec2_instance:
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"
        region: "{{ region }}"
        instance_ids: "{{ instance_ids }}"
        state: absent
    # Delete Security Group
    - name: Delete Security Group
      ec2_group:
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"
        region: "{{ region }}"
        name: "{{ security_group_name }}"
        group_id: "{{ security_group_id }}"
"delete1.yaml" 102L, 3242B
17,35 Top
27°C Haze Search ENG IN 18:19 05-12-2024
```

```
ec2-user@ip-172-31-29-229:~ % Windows PowerShell
#
# Delete Security Group
- name: Delete Security Group
  ec2_group:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    name: "{{ security_group_name }}"
    group_id: "{{ security_group_id }}"
    state: absent
#
# Delete Public Subnet
- name: Delete Public Subnet
  ec2_vpc_subnet:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    cidr: "{{ pubsubnet_cidr_block }}"
    state: absent
#
# Delete Private Subnet
- name: Delete Private Subnet
  ec2_vpc_subnet:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    cidr: "{{ pvtsubnet_cidr_block }}"
    state: absent
#
# Delete Public Route Table
- name: Delete Public Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    tags:
      Name: "{{ pubroute_table_name }}"
    state: absent
67,39 51%
27°C Haze Search ENG IN 18:19 05-12-2024
```



```

# Delete Public Route Table
- name: Delete Public Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    tags:
      Name: "{{ pubroute_table_name }}"
    state: absent

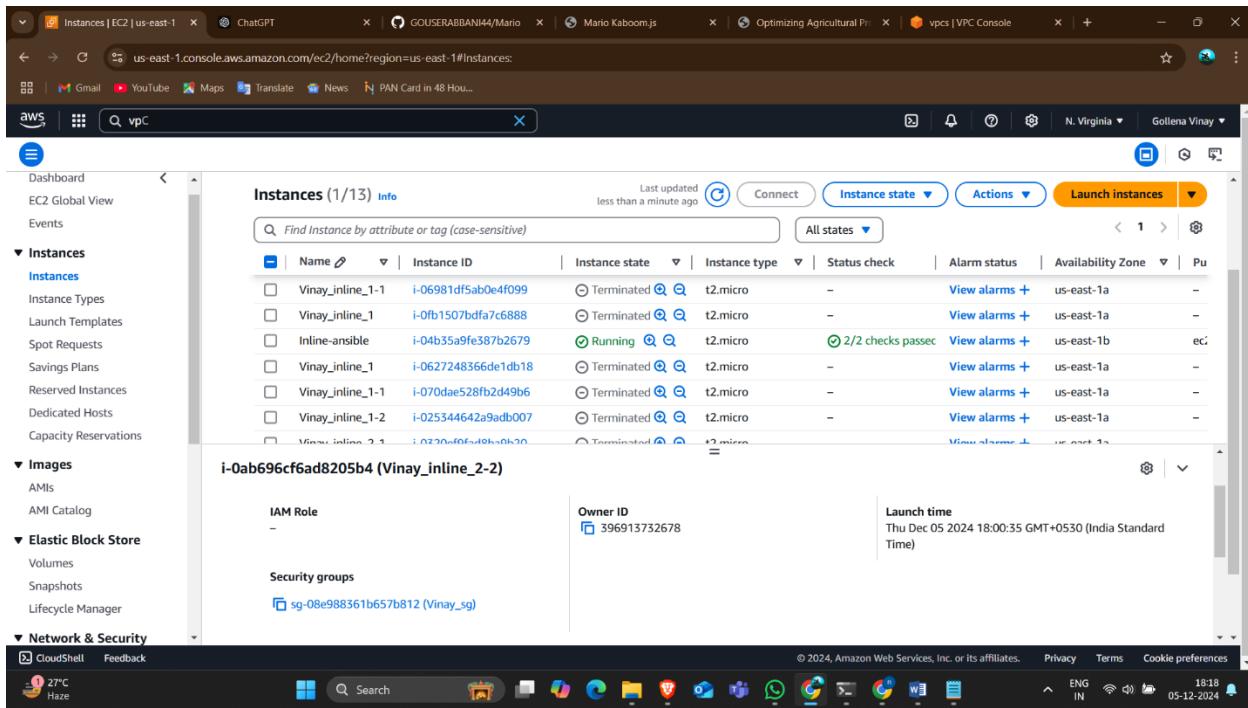
# Delete Private Route Table
- name: Delete Private Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    tags:
      Name: "{{ pvrouting_table_name }}"
    state: absent

# Detach and Delete Internet Gateway
- name: Detach and Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "{{ vpc_id }}"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

# Delete VPC
- name: Delete VPC by name
  ec2_vpc_net:
    cidr_block: "11.0.0.0/16" # Replace with your VPC CIDR block
    name: "{{ vpc_name }}"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

```

- Here the result of deleting the resources.



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Vinay_inline_1-1	i-06981df5ab0e4f099	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a
Vinay_inline_1	i-0fb1507bd7a7c6888	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a
Inline-ansible	i-04b35a9fe387b2679	Running	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1b
Vinay_inline_1	i-0627248366de1db18	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a
Vinay_inline_1-1	i-070dae528fb2d49b6	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a
Vinay_inline_1-2	i-02534462a9adb007	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a
Vinay_inline_2-1	i-0270aef610b80b20	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a

**i-0ab696cf6ad8205b4 (Vinay\_inline\_2-2)**

IAM Role: -

Owner ID: 396913732678

Launch time: Thu Dec 05 2024 18:00:35 GMT+0530 (India Standard Time)

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with navigation links for EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections), and Security (Network ACLs). The main area displays 'Your VPCs (1) Info' with a table showing one VPC entry:

Name	VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
-	vpc-085ec71d994a9a322	Available	Off	172.31.0.0/16	-

Below the table, a detailed view for 'vpc-0f2b11cf84a5c5022 / Vinay-vpc' is shown with tabs for Details, Resource map, CIDs, Flow logs, Tags, and Integrations. The Details tab shows the following configuration:

VPC ID	State	Block Public Access	DNS hostnames
vpc-0f2b11cf84a5c5022	Available	Off	Enabled
DNS resolution	Tenancy	DHCP option set	Main route table
-	-	-	-

At the bottom, there are links for CloudShell, Feedback, and various AWS services like Lambda, S3, CloudWatch, etc.

```
# Delete Private Route Table
- name: Delete Private Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "{{ vpc_id }}"
    tags:
      Name: "{{ pvtroute_table_name }}"
    state: absent

# Detach and Delete Internet Gateway
- name: Detach and Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "{{ vpc_id }}"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

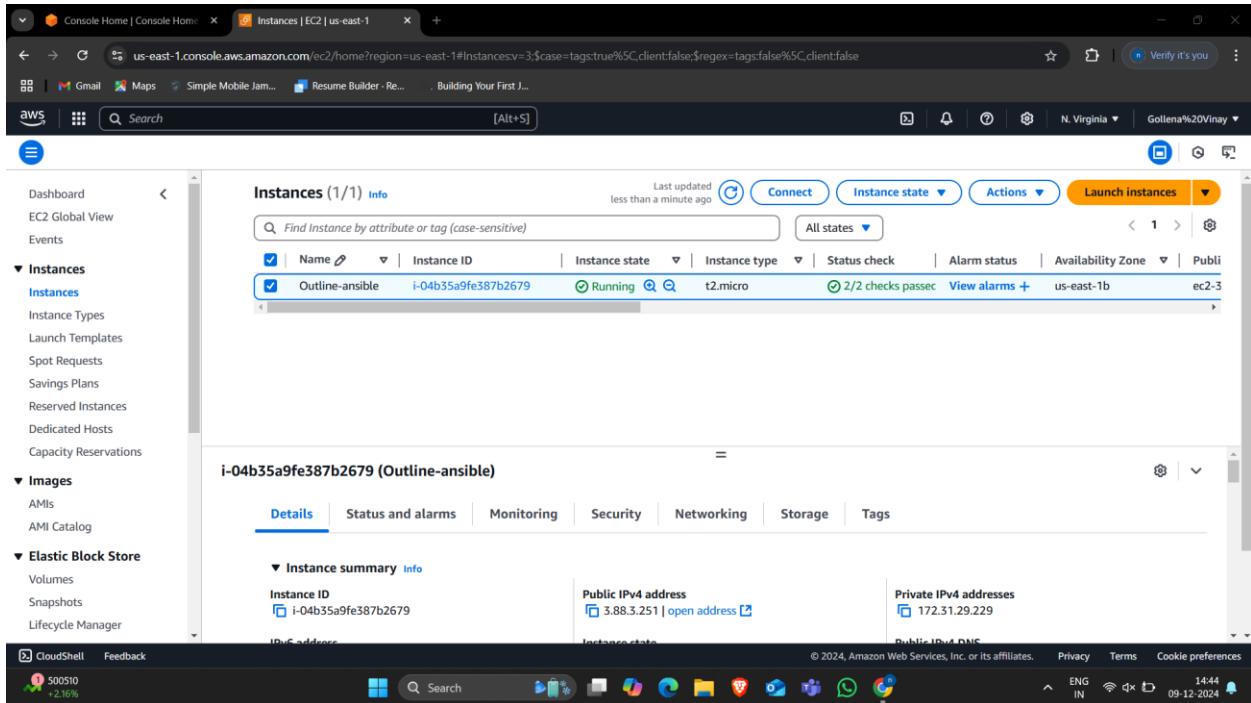
# Delete VPC
- name: Delete VPC by name
  ec2_vpc_net:
    cidr_block: "11.0.0.0/16" # Replace with your VPC CIDR block
    name: "{{ vpc_name }}"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
[ec2-user@ip-172-31-29-229 ~]$ sudo vi static.sh
[ec2-user@ip-172-31-29-229 ~]$ sudo vi dynamic.sh
[ec2-user@ip-172-31-29-229 ~]$ ll
total 36
-rw-r--r-- 1 root      root      3242 Dec  5 12:46 delete.yaml
-rw-r--r-- 1 root      root      2589 Dec  5 11:55 delete.yaml
-rw-r--r-- 1 root      root      193 Dec  5 12:17 dynamic.sh
-rw-r--r-- 1 root      root      4593 Dec  5 12:30 inline.yaml
-rw-r--r-- 1 root      root      4861 Dec  5 11:11 Inline.yaml
-rw-r--r-- 1 root      root      207 Dec  5 12:35 static.sh
-rw----- 1 ec2-user  ec2-user   1678 Dec  5 08:52 vinay.pem
[ec2-user@ip-172-31-29-229 ~]$ |
```

The screenshot shows a Windows PowerShell window with the title 'Windows PowerShell'. It contains a command history for deleting a VPC using Ansible. The commands involve deleting a route table, detaching and deleting an internet gateway, and deleting the VPC itself. The output shows the creation of static and dynamic configuration files ('static.sh', 'dynamic.sh') and the final command being run ('ll'). The PowerShell window has a standard Windows taskbar at the bottom with icons for File Explorer, Task View, Start, and other system tools.

## Method-3 (Outline Variables)

- ❖ Create the resources by using the outline variable for playbook.

- Launch an instance by using amazon-linux.



- Connect to the terminal.update it.
- Than copy the pem file and give permission.
- Change the hosts file and configure file of ansible.
- Go to usr/bin/ than install the boto boto3 botocore.
- Write the userdata files,
- Write the outline variable file and name it “secret.yaml”
- Write the playbook file and execute.

```

[ec2-user@ip-172-31-29-229:~] ok: [Inline-ansible]
TASK [Create a Security Group] *****
ok: [Inline-ansible]
TASK [Launch EC2 Instance] *****
changed: [Inline-ansible]
TASK [Launch EC2 Instance] *****
changed: [Inline-ansible]
PLAY RECAP *****
Inline-ansible : ok=9    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-29-229 ~]$ sudo vi vinay.yaml
[ec2-user@ip-172-31-29-229 ~]$ client_loop: send disconnect: Connection reset
C:\Users\user\Downloads\ - i "vinay.pem" ec2-user@ec2-3-88-3-251.compute-1.amazonaws.com
Last login: Mon Dec 9 11:37:08 2024 from ip-172-31-29-229.ec2.internal
#
# Amazon Linux 2
## AL2 End of Life is 2025-06-30.
## A newer version of Amazon Linux is available!
## Amazon Linux 2023, GA and supported until 2028-03-15.
## https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-29-229 ~]$ cat dynamic.sh
#!/bin/bash
sudo yum install git -y
sudo git clone https://github.com/vinayl8-2/flight-perdiction.git
sudo mv flight-perdiction /home/ec2-user/
cd /home/ec2-user/flight-perdiction
pip3 install -r requirements.txt
screen -m -d python3 app.py
[ec2-user@ip-172-31-29-229 ~]$ |
```

Breaking news AAP 2nd candid... 17:15 09-12-2024

- Here the Outline variable file “secret.yaml”

```

# AWS Credentials (ensure to store these securely)
aws_access_key: "AKIAVY2PGSRD05KUAWCJ"
aws_secret_key: "fh9F0UPifZiu9zdx0DQczJXMIGWFN9y+nik5E5Gk"
title: "demo"

# VPC Configuration
vpc_name: "Vinay_vpc"
vpc_cidr_block: "11.0.0.0/16"
region: "us-east-1"

# Internet Gateway Configuration
igw_name: "Vinay_igw"

# Public Subnet Configuration
pubsubnet_name: "Vinay_pub"
pubsubnet_cidr_block: "11.0.1.0/24"
pubzone: "us-east-1a"

# Private Subnet Configuration
pvtsubnet_name: "Vinay_pvt"
pvtsubnet_cidr_block: "11.0.2.0/24"
pvtzone: "us-east-1b"

# Public Route Table Configuration
pubroute_table_name: "Vinay_pub_rt"

# Private Route Table Configuration
pvtroute_table_name: "Vinay_pvt_rt"

# Security Group Configuration
security_group_name: "Vinay_sg"
destination_cidr_block: "0.0.0.0/0"
port22_cidr_block: "0.0.0.0/0"

# EC2 Instance Configuration
image_id: "ami-0166fe664262f664c"
type: "t2.micro"
instance_name1: "Vinay-static"
instance_name2: "Vinay-dynamic"
~
```

"secret.yaml" 39L, 1029B 38,31 All

BSE midcap +0.41% 17:58 09-12-2024

- Here the ansible.yaml.

```
ec2-user@ip-172-31-29-229:~ % + v
- hosts: localhost
  become: yes
  gather_facts: false

  vars_files:
    - secret.yaml

  tasks:
    ## VPC Creation ##
    - name: Create a VPC
      ec2_vpc_net:
        aws_access_key: "{{ aws_access_key }}"
        aws_secret_key: "{{ aws_secret_key }}"
        cidr_block: "{{ vpc_cidr_block }}"
        name: "{{ vpc_name }}"
        region: "{{ region }}"
        dns_support: yes # Enable DNS support
        dns_hostnames: yes # Enable DNS hostnames
        tenancy: default
        state: present
      register: vpc_result

    ## Internet Gateway Creation ##
    - name: Create an Internet Gateway
      ec2_vpc_igw:
        aws_access_key: "{{ aws.access_key }}"
        aws_secret_key: "{{ aws.secret_key }}"
        vpc_id: "{{ vpc_result.vpc.id }}"
        region: "{{ region }}"
        state: present
        tags:
          Name: "{{ igw_name }}"
      register: igw_result

    ## Public Subnet Creation ##
    - name: Create a Public Subnet
      ec2_vpc_subnet:
        aws_access_key: "{{ aws.access_key }}"
        aws_secret_key: "{{ aws.secret_key }}"
        vpc_id: "{{ vpc_result.vpc.id }}"
    -- INSERT --
1,19           Top

```

Wi - BAN Game score

Search

17:17 09-12-2024

```
ec2-user@ip-172-31-29-229:~ % + v
    ## Public Subnet Creation ##
    - name: Create a Public Subnet
      ec2_vpc_subnet:
        aws_access_key: "{{ aws.access_key }}"
        aws_secret_key: "{{ aws.secret_key }}"
        vpc_id: "{{ vpc_result.vpc.id }}"
        region: "{{ region }}"
        az: "{{ pubzone }}" # Specify the Availability Zone
        state: present
        cidr: "{{ pubsubnet_cidr_block }}"
        map_public: yes # Enable automatic public IP assignment
        resource_tags:
          Name: "{{ pubsubnet_name }}"
      register: pubsubnet_result

    ## Private Subnet Creation ##
    - name: Create a Private Subnet
      ec2_vpc_subnet:
        aws_access_key: "{{ aws.access_key }}"
        aws_secret_key: "{{ aws.secret_key }}"
        vpc_id: "{{ vpc_result.vpc.id }}"
        region: "{{ region }}"
        az: "{{ pvtzone }}" # Specify the Availability Zone
        state: present
        cidr: "{{ pvtsubnet_cidr_block }}"
        map_public: no # Disable automatic public IP assignment
        resource_tags:
          Name: "{{ pvtsubnet_name }}"
      register: pvtsubnet_result

    ## Public Route Table ##
    - name: Create a Public Route Table
      ec2_vpc_route_table:
        aws_access_key: "{{ aws.access_key }}"
        aws_secret_key: "{{ aws.secret_key }}"
        vpc_id: "{{ vpc_result.vpc.id }}"
        region: "{{ region }}"
        state: present
        tags:
    -- INSERT --
68,35           30%

```

Wi - BAN Game score

Search

17:17 09-12-2024

```
ec2-user@ip-172-31-29-229:~ % + ^

### Public Route Table ###
- name: Create a Public Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: "{{ region }}"
    state: present
    tags:
      Name: "{{ pubroute_table_name }}"
    subnets:
      - "{{ pubsubnet_result.subnet.id }}"
    routes:
      - dest: "0.0.0.0/0" # Default route to the internet
        gateway_id: "{{ igw_result.gateway_id }}"
  register: public_route_table

### Private Route Table ###
- name: Create a Private Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: "{{ region }}"
    state: present
    tags:
      Name: "{{ pvtoute_table_name }}"
    subnets:
      - "{{ pvtsubnet_result.subnet.id }}"
  register: private_route_table

### Security Group Creation ###
- name: Create a Security Group
  ec2_group:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: "{{ region }}"
    state: present
    name: "{{ security_group_name }}"
-- INSERT --
99, 35      58%
```



```
ec2-user@ip-172-31-29-229:~ % + ^

### Security Group Creation ###
- name: Create a Security Group
  ec2_group:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    vpc_id: "{{ vpc_result.vpc.id }}"
    region: "{{ region }}"
    state: present
    name: "{{ security_group_name }}"
    description: "Allow all traffic" # A description of the security group
    tags:
      Name: "vinay-sg"
    rules:
      - proto: all
        cidr_ip: "0.0.0.0/0"
        rule_desc: "Allow all traffic"
  register: security_group_results

### EC2 Instance Launch ###
- name: Launch EC2 Instance
  ec2:
    image: "ami-0166fe664262f664c"
    instance_type: "{{ type }}"
    region: "{{ region }}"
    wait: yes
    count: 1
    state: present
    vpc_subnet_id: "{{ pubsubnet_result.subnet.id }}"
    assign_public_ip: yes
    group_id: "{{ security_group_results.group_id }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    user_data: "{{ lookup('file', 'static.sh') }}"
    instance_tags:
      Name: "{{ instance_name1 }}"
  register: ec2_instance_result

- name: Launch EC2 Instance
  ec2:
-- INSERT --
129, 23      86%
```



```

ec2-user@ip-172-31-29-22 ~ + 
    cidr_ip: "0.0.0.0/0"
    rule_desc: "Allow all traffic"
register: security_group_results

### EC2 Instance Launch ###
- name: Launch EC2 Instance
  ec2:
    image: "ami-0166fe664262f664c"
    instance_type: "{{ type }}"
    region: "{{ region }}"
    wait: yes
    count: 1
    state: present
    vpc_subnet_id: "{{ pubsubnet_result.subnet_id }}"
    assign_public_ip: yes
    group_id: "{{ security_group_results.group_id }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    user_data: "{{ lookup('file', 'static.sh') }}"
    instance_tags:
      Name: "{{ instance_name }}"
register: ec2_instance_result

- name: Launch EC2 Instance
  ec2:
    image: "ami-0166fe664262f664c"
    instance_type: "{{ type }}"
    region: "{{ region }}"
    wait: yes
    count: 1
    state: present
    vpc_subnet_id: "{{ pubsubnet_result.subnet_id }}"
    assign_public_ip: yes
    group_id: "{{ security_group_results.group_id }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    user_data: "{{ lookup('file', 'dynamic.sh') }}"
    instance_tags:
      Name: "{{ instance_name2 }}"
register: ec2_instance_result
-- INSERT --

```

149,35 Bot

WI-BAN ENG IN 17:17 09-12-2024

- Execute the playbook file.

```

ec2-user@ip-172-31-29-229:~ + 
[ec2-user@ip-172-31-29-229 ~]$ "vinay.yaml" 149L, 4849B written
[ec2-user@ip-172-31-29-229 ~]$ sudo vi static.sh
[ec2-user@ip-172-31-29-229 ~]$ "static.sh" 8L, 2088B written
[ec2-user@ip-172-31-29-229 ~]$ sudo vi dynamic.sh
[ec2-user@ip-172-31-29-229 ~]$ 7L, 245B written
[ec2-user@ip-172-31-29-229 ~]$ ansible-playbook vinay.yaml

PLAY [localhost] ****
TASK [Create a VPC] ****
ok: [Inline-ansible]
TASK [Create an Internet Gateway] ****
ok: [Inline-ansible]
TASK [Create a Public Subnet] ****
ok: [Inline-ansible]
TASK [Create a Private Subnet] ****
ok: [Inline-ansible]
TASK [Create a Public Route Table] ****
ok: [Inline-ansible]
TASK [Create a Private Route Table] ****
ok: [Inline-ansible]
TASK [Create a Security Group] ****
ok: [Inline-ansible]
TASK [Launch EC2 Instance] ****
changed: [Inline-ansible]
TASK [Launch EC2 Instance] ****
changed: [Inline-ansible]

PLAY RECAP ****
  : ok=9  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

[ec2-user@ip-172-31-29-229 ~]$ sudo vi vinay.yaml
[ec2-user@ip-172-31-29-229 ~]$ client_loop: send disconnect: Connection reset

```

27°C Haze ENG IN 18:03 09-12-2024

- Here we can see the data files.

```
[ec2-user@ip-172-31-29-229 ~]$ + ^

TASK [Launch EC2 Instance] *****
changed: [Inline-ansible]

PLAY RECAP *****
Inline-ansible : ok=9    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[ec2-user@ip-172-31-29-229 ~]$ sudo vi vinay.yaml
[ec2-user@ip-172-31-29-229 ~]$ client_loop; send disconnect: Connection reset

C:\Users\user\Downloads>ssh -i "vinay.pem" ec2-user@ec2-3-88-3-251.compute-1.amazonaws.com
Last login: Mon Dec  9 11:37:08 2024 from ip-172-31-29-229.ec2.internal
#
#_ _ _ _ _ Amazon Linux 2
#_ _ \_ _ _ # AL2 End of Life is 2025-06-30.
#_ _ \# / _ _ _ A newer version of Amazon Linux is available!
#_ _ \_ / _ _ / Amazon Linux 2023, GA and supported until 2028-03-15.
#_ /m/ https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-29-229 ~]$ cat dynamic.sh
#!/bin/bash
sudo yum install git -y
sudo git clone https://github.com/vinay18-2/flight-perdiction.git
sudo mv flight-perdiction/ /home/ec2-user/
cd /home/ec2-user/Flight-perdiction
pip3 install -r requirements.txt
screen -m -d python3 app.py
[ec2-user@ip-172-31-29-229 ~]$ cat static.sh
#!/bin/bash
sudo yum install -y httpd
sudo systemctl start httpd
sudo systemctl enable httpd
sudo yum install git -y
sudo git clone https://github.com/Gouserabbani44/Mario.git
sudo mv Mario/* /var/www/html/
[ec2-user@ip-172-31-29-229 ~]$ |
```

- We can observe the created instances.

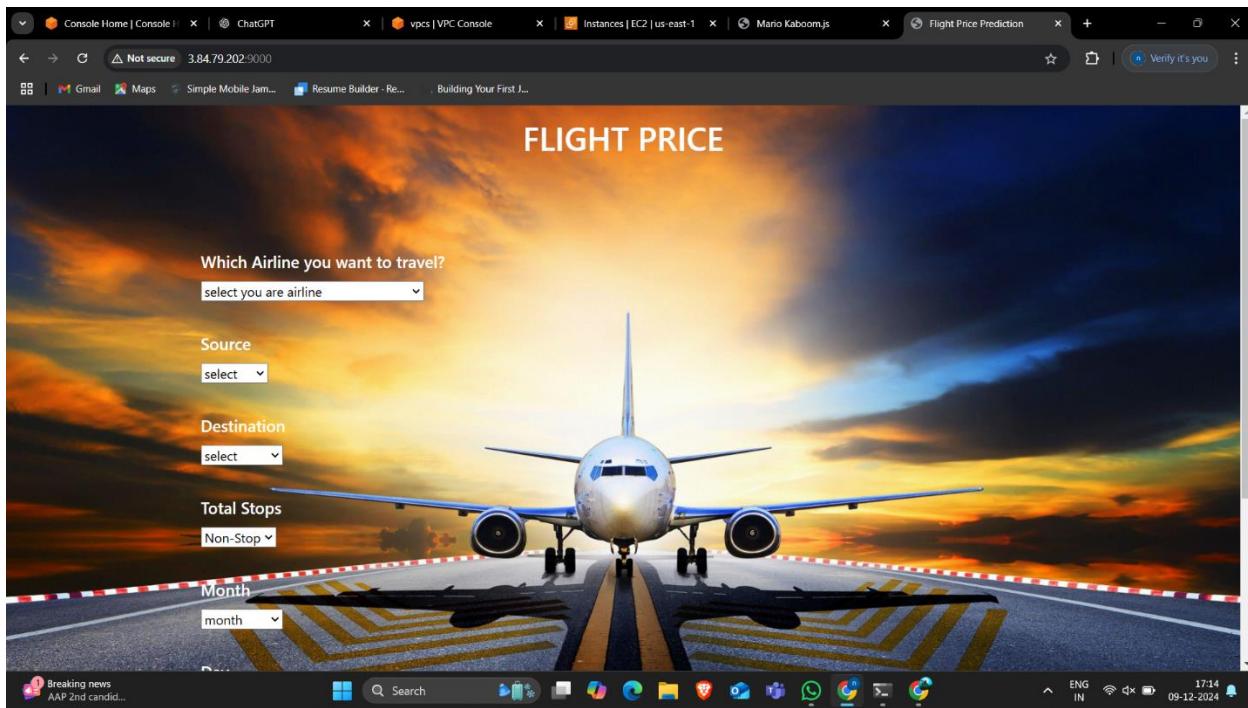
The screenshot shows the AWS Management Console with the Instances page open. The main content area displays a table of three EC2 instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
Vinay-dynamic	i-0efd010f5e68daec6	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec
Vinay-static	i-0d358bd0b0392d0dd	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec
Outline-ansible	i-04b35a9fe587b2679	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec

Below the table, a detailed view for the selected instance (i-0efd010f5e68daec6) is shown. The Details tab is active, displaying the following information:

- Instance summary:** Public IPv4 address: 3.84.79.202 | open address, Instance state: Running.
- Public IPv4 DNS:** ec2-3-84-79-202.compute-1.amazonaws.com | open address.
- Private IPv4 addresses:** 11.0.1.253.

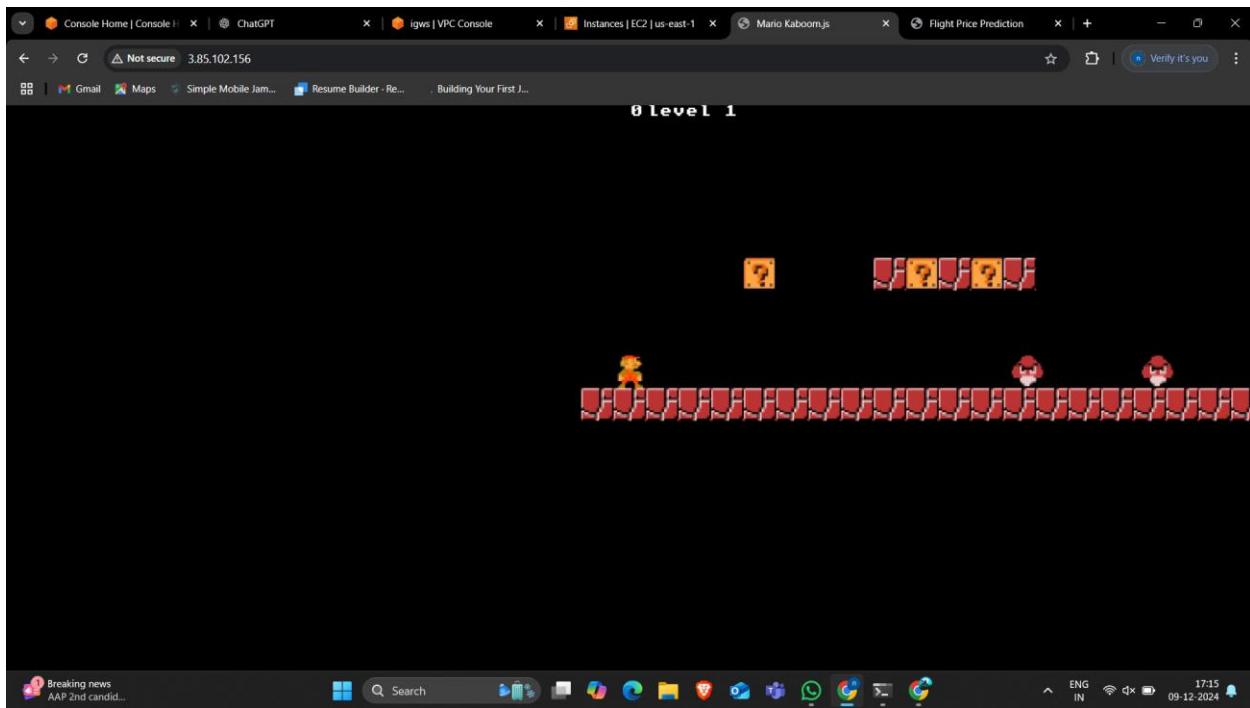
- Result of dynamic application.



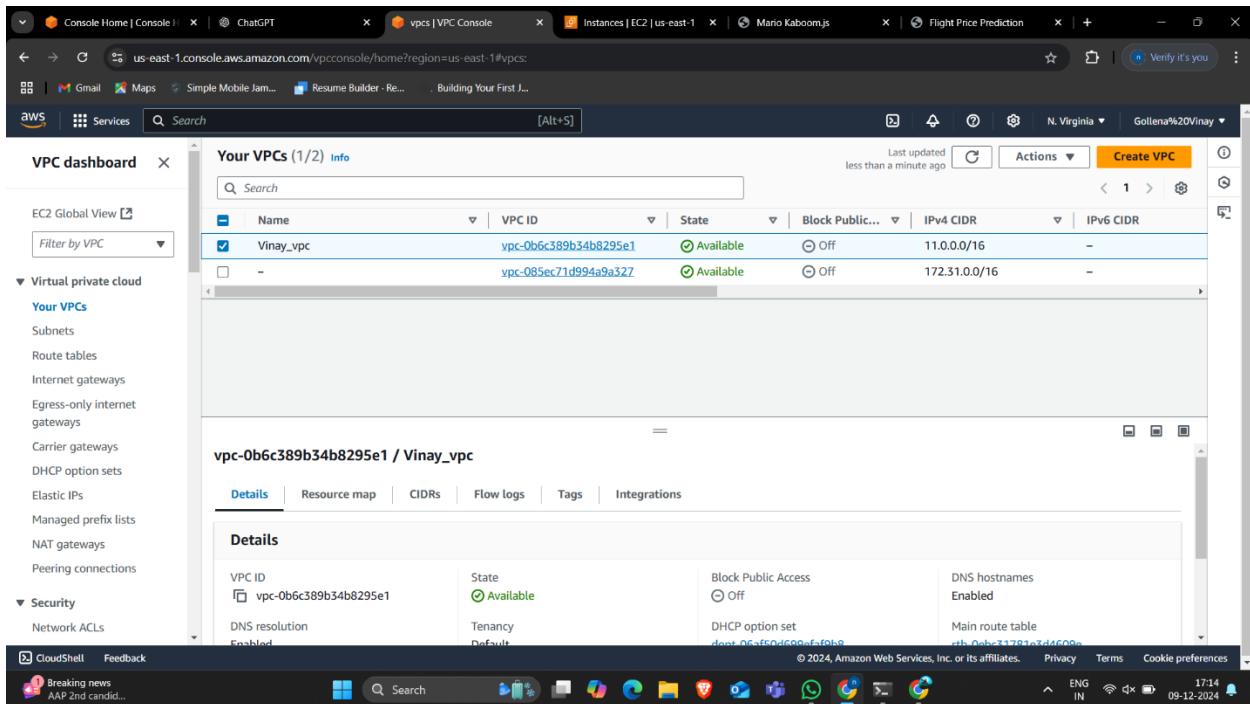
- Instance-2

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
Vinay-dynamic	i-0efd010f5e68daec6	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
<b>Vinay-static</b>	<b>i-0d358bd0b0392d0dd</b>	<b>Running</b>	<b>t2.micro</b>	<b>2/2 checks passed</b>	<b>View alarms +</b>	<b>us-east-1a</b>	<b>ec2</b>
Outline-ansible	i-04b35a9fe387b2679	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2

- Result of static application.



- Result of vpc created.



- Subnets.

**Subnets (2/8) Info**

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0350327d6f7894f1b	Available	vpc-085ec71d994a9a327	Off	172.31.80.0/24
<input checked="" type="checkbox"/> Vinay_pub	subnet-079a4b9b2d5a7c76c	Available	vpc-0b6c389b34b8295e1   Vinay...	Off	11.0.1.0/24
-	subnet-0ecbf1f3fab4a01c7	Available	vpc-085ec71d994a9a327	Off	172.31.32.0/24
-	subnet-09cbb0d092842ce23	Available	vpc-085ec71d994a9a327	Off	172.31.0.0/24
-	subnet-0fad19b11702ec191	Available	vpc-085ec71d994a9a327	Off	172.31.64.0/24
-	subnet-040b9c0e422606bfaf	Available	vpc-085ec71d994a9a327	Off	172.31.48.0/24
-	subnet-0f9874237f473da88	Available	vpc-085ec71d994a9a327	Off	172.31.16.0/24
<input checked="" type="checkbox"/> Vinay_pvt	subnet-0ad8cbd07f73d4f29	Available	vpc-0b6c389b34b8295e1   Vinay...	Off	11.0.2.0/24

Subnets: subnet-0ad8cbd07f73d4f29, subnet-079a4b9b2d5a7c76c

- Route tables.

**Route tables (2/4) Info**

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
<input checked="" type="checkbox"/> Vinay_pub_rt	rtb-041605e76f7d44f70	subnet-079a4b9b2d5a7c...	-	No	vpc-0b6c389b34b8295e1
-	rtb-08d790e48086a2fb	-	-	Yes	vpc-085ec71d994a9a327
-	rtb-0ebc31781e3d4609e	-	-	Yes	vpc-0b6c389b34b8295e1
<input checked="" type="checkbox"/> Vinay_pvt_rt	rtb-0778a45ed19319bca	subnet-0ad8cbd07f73d4f...	-	No	vpc-0b6c389b34b8295e1

Route tables: rtb-041605e76f7d44f70, rtb-0778a45ed19319bca

- Internet gateway.

Name	Internet gateway ID	State	VPC ID	Owner
Vinay_igw	igw-06d8b4cbc04690b46	Attached	vpc-0b6c389b34b8295e1   Vinay_vpc	396913732678
-	igw-0fe9e5fd24c77548	Attached	vpc-085ec71d994a9a327	396913732678

**igw-06d8b4cbc04690b46 / Vinay\_igw**

**Details** | Tags

**Details**

Internet gateway ID: igw-06d8b4cbc04690b46 | State: Attached | VPC ID: vpc-0b6c389b34b8295e1 | Owner: 396913732678

- Here playbook script for deletion.

```

hosts: localhost
become: yes
gather_facts: false

vars_files:
- secret.yaml
tasks:
# Terminate EC2 Instances
- name: Terminate EC2 Instance
  ec2_instance:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    instance_ids:
      - "i-0efd01f5e68dsec6"
      - "i-0d358bd0b0392d0dd"
    state: absent

# Delete Security Group
- name: Delete Security Group
  ec2_group:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    name: "{{ security_group_name }}"
    group_id: "sg-026c5b0737e25ca6f"
    state: absent

# Delete Public Subnet
- name: Delete Public Subnet
  ec2_vpc_subnet:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    cidr: "subnet-079a4b9b2d5a7c76c"
    state: absent

# Delete Private Subnet
- name: Delete Private Subnet

```

```
# Delete Private Subnet
- name: Delete Private Subnet
  ec2_vpc_subnet:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    cidr: "subnet-0ad8cbd07f73d4f29"
    state: absent

# Delete Public Route Table
- name: Delete Public Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    tags:
      Name: "{{ pubroute_table_name }}"
    state: absent

# Delete Private Route Table
- name: Delete Private Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    tags:
      Name: "{{ pvrouting_table_name }}"
    state: absent

# Detach and Delete Internet Gateway
- name: Detach and Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "vpc-0b6c389b34b8295e1"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"

43,36 77%
```

```
# Delete Public Route Table
- name: Delete Public Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    tags:
      Name: "{{ pubroute_table_name }}"
    state: absent

# Delete Private Route Table
- name: Delete Private Route Table
  ec2_vpc_route_table:
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"
    region: "{{ region }}"
    vpc_id: "vpc-0b6c389b34b8295e1"
    tags:
      Name: "{{ pvrouting_table_name }}"
    state: absent

# Detach and Delete Internet Gateway
- name: Detach and Delete Internet Gateway
  ec2_vpc_igw:
    vpc_id: "vpc-0b6c389b34b8295e1"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

# Delete VPC
- name: Delete VPC by name
  ec2_vpc_net:
    cidr_block: "11.0.0.0/16" # Replace with your VPC CIDR block
    name: "vpc-0b6c389b34b8295e1"
    state: absent
    region: "{{ region }}"
    aws_access_key: "{{ aws_access_key }}"
    aws_secret_key: "{{ aws_secret_key }}"

88,36 Bot
```

- Execute the deletion script file.

```

ec2-user@ip-172-31-29-229:~ % cat OTdlt.yaml
PLAY RECAP ****
[ec2-user@ip-172-31-29-229 ~]$ sudo vi OTdlt.yaml
[ec2-user@ip-172-31-29-229 ~]$ "OTdlt.yaml" 88L, 2660B written
[ec2-user@ip-172-31-29-229 ~]$ ansible-playbook OTdlt.yaml

PLAY [localhost] ****
TASK [Terminate EC2 Instance] ****
ok: [Inline-ansible]

TASK [Delete Security Group] ****
ok: [Inline-ansible]

TASK [Delete Public Subnet] ****
ok: [Inline-ansible]

TASK [Delete Private Subnet] ****
ok: [Inline-ansible]

TASK [Delete Public Route Table] ****
ok: [Inline-ansible]

TASK [Delete Private Route Table] ****
ok: [Inline-ansible]

TASK [Detach and Delete Internet Gateway] ****
ok: [Inline-ansible]

TASK [Delete VPC by name] ****
ok: [Inline-ansible]

PLAY RECAP ****
ok: [Inline-ansible]
[ec2-user@ip-172-31-29-229 ~]$ cat OTdlt.yaml
- hosts: localhost
  become: yes

```

27°C Haze

Search

ENG IN 18:03 09-12-2024

- Observe the deleted resources.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pu
Vinay-dynamic	i-0efd010f5e68daec6	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a	-
<b>Vinay-static</b>	<b>i-0d358bd0b0392d0dd</b>	Terminated	t2.micro	-	<a href="#">View alarms +</a>	us-east-1a	-
Outline-ansible	i-04b35a9fe387b2679	Running	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1b	ec2

**i-0d358bd0b0392d0dd (Vinay-static)**

**Security details**

IAM Role: -

Owner ID: 396913732678

Launch time: Mon Dec 09 2024 17:07:01 GMT+0530 (India Standard Time)

**Inbound rules**

CloudShell Feedback

27°C Haze

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ENG IN 18:03 09-12-2024