Ansible-Terraform-Project

Prerequisite:

- Terraform installed

```
curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key
sudo apt-add-repository "deb [arch=amd64] https://apt.releases.h
sudo apt-get update && sudo apt-get install terraform
```

- aws-cli installed

```
sudo apt install awscli -y
```

Steps:

- **Install ansible in the system**

```
sudo apt-add-repository ppa:ansible/ansible
sudo apt-get update
sudo apt install ansible -y
ansible  --version
```

- **We need an ec2 instance where we will install and run nginx using ansible**

  → Clone github repo for ec2 creating using terraform (we have worked on it in earlier projects)

  Github repo: https://github.com/SachinR007/terraform-ec2

→ To create ec2 instance using terraform, clone below repository (we worked on it in previous project)

Github repo: https://github.com/SachinR007/terraform-ec2

→ Go inside the cloned repository and run `terraform init` to load the required aws provider.

```
ubuntu@ip-172-31-80-135:~$ ls
bin  install.sh  terraform-ec2
ubuntu@ip-172-31-80-135:~$ cd terraform-ec2/
ubuntu@ip-172-31-80-135:~/terraform-ec2$
ubuntu@ip-172-31-80-135:~/terraform-ec2$
ubuntu@ip-172-31-80-135:~/terraform-ec2$
ubuntu@ip-172-31-80-135:~/terraform-ec2$ terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.40.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
ubuntu@ip-172-31-80-135:~/terraform-ec2$
```

→ update variables.tf file with your aws credentials from a role which has access to create ec2 instance

→ Run `terraform plan` and check if it is fine then run `terraform apply --auto-approve`

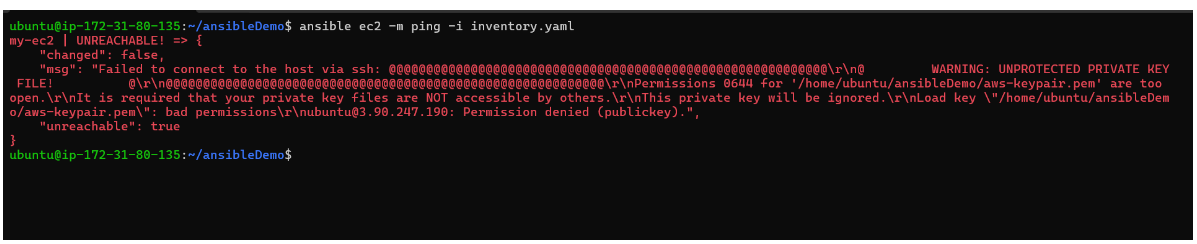
```
Plan: 2 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + display  = (known after apply)
  + display_ = (known after apply)
aws_security_group.allow_tls: Creating...
aws_security_group.allow_tls: Creation complete after 2s [id=sg-01b704e4c0657767f]
aws_instance.aws_test: Creating...
aws_instance.aws_test: Still creating... [10s elapsed]
aws_instance.aws_test: Still creating... [20s elapsed]
aws_instance.aws_test: Still creating... [30s elapsed]
aws_instance.aws_test: Creation complete after 32s [id=i-0505c2503d0110eaa]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

display = "3.90.247.190"
display_ = toset([
  "Secgrp_terraform",
])
ubuntu@ip-172-31-80-135:~/terraform-ec2$
```

> 💡 Here we got the IP of new instance, store it somewhere it will be required later for ansible.

- Create an inventory file for ansible having the IP of ec2 created by terraform.

```
#inventory.yaml
ec2:
  hosts:
   my-ec2:
    ansible_host: <Your ip here> # add your IP here
    ansible_user: ubuntu # change user if you are not using u
    ansible_ssh_private_key_file: /home/ubuntu/ansibleDemo/aw
            # give your keypair info in this key
```

- Lets try pinging our newly created ec2 instance using `ansible ec2 -m ping -i inventory.yaml`



Oh no!! we got an error... Ahh well, this is life of a devops engineer😁 We will definitely see errors, but we have to keep trying to fix it and make it work...

Here It says key is too open/ bad permissions. We can fix it by reducing the permissions of the pem file.

```
ubuntu@ip-172-31-80-135:~/ansibleDemo$ ls -lrt
total 8
-rw-rw-r-- 1 ubuntu ubuntu  156 Mar 13 15:23 inventory.yaml
-rw-r--r-- 1 ubuntu ubuntu 1678 Mar 13 15:34 aws-keypair.pem
ubuntu@ip-172-31-80-135:~/ansibleDemo$ chmod 400 aws-keypair.pem
ubuntu@ip-172-31-80-135:~/ansibleDemo$
ubuntu@ip-172-31-80-135:~/ansibleDemo$ ls -lrt
total 8
-rw-rw-r-- 1 ubuntu ubuntu  156 Mar 13 15:23 inventory.yaml
-r-------- 1 ubuntu ubuntu 1678 Mar 13 15:34 aws-keypair.pem
ubuntu@ip-172-31-80-135:~/ansibleDemo$
```

Yay!! we have successfully cleared one hurdle of life.. We can ping our ec2 instance now.

```
ubuntu@ip-172-31-80-135:~/ansibleDemo$
ubuntu@ip-172-31-80-135:~/ansibleDemo$ ansible ec2 -m ping -i inventory.yaml
my-ec2 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-80-135:~/ansibleDemo$
```

- Create a basic `index.html` file

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial
  <title>Ansible Success!</title>
  <style>
    body {
      background-color: #ff9999; /* Light salmon pink */
      color: #333333; /* Dark gray */
      font-family: Arial, sans-serif;
```

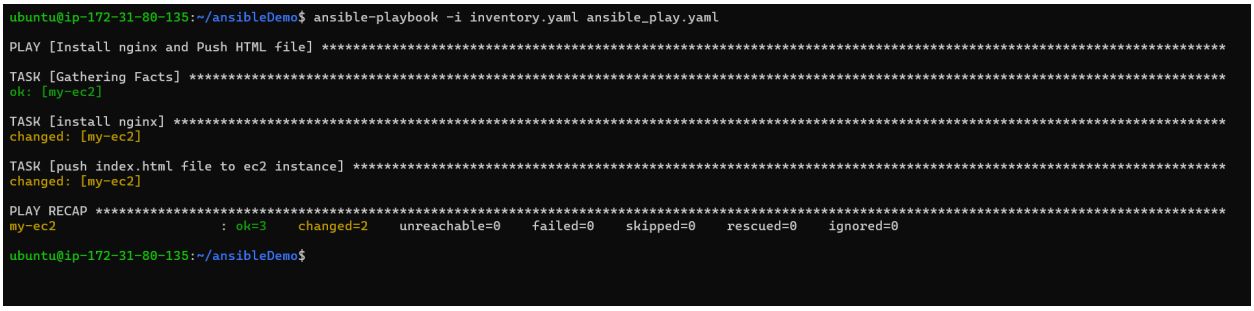
```
      margin: 0;
      padding: 20px;
      text-align: center;
    }
    h1 {
      font-size: 4em;
      margin: 20px 0;
    }
  </style>
</head>
<body>
  <h1>YAY!!!</h1>
  <p>We have learnt Ansible!</p>
</body>
</html>
```

- create a playbook to install nginx and push the index.html file in ec2 instance.

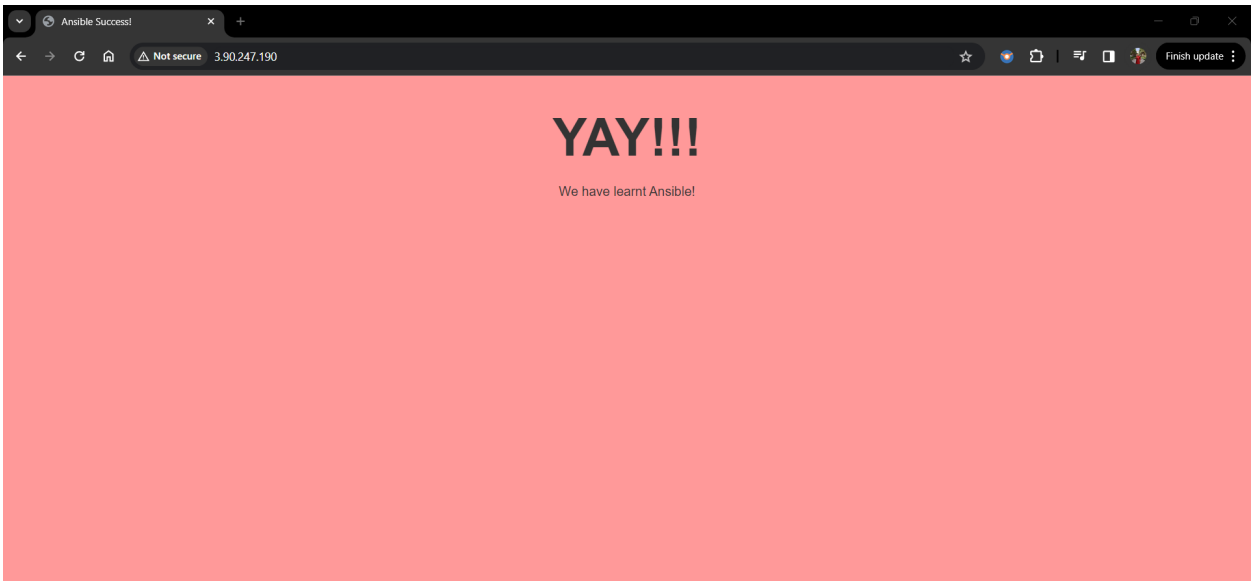
```
---
- name: Install nginx
  hosts: ec2
  become: yes
  tasks:
   - name: install nginx
     apt:
      name: nginx
      state: latest
      update_cache: yes

   - name: push index.html file to ec2 instance
     copy:
      src: /home/ubuntu/ansibleDemo/index.html
      dest: /var/www/html/index.html
```

```
ubuntu@ip-172-31-80-135:~/ansibleDemo$ cat ansible_play.yaml
---
- name: Install nginx and Push HTML file
  hosts: ec2
  become: yes
  tasks:
   - name: install nginx
     apt:
       name: nginx
       state: latest
       update_cache: yes

   - name: push index.html file to ec2 instance
     copy:
       src: /home/ubuntu/ansibleDemo/index.html
       dest: /var/www/html/index.html
ubuntu@ip-172-31-80-135:~/ansibleDemo$
```

- Run `ansible-playbook -i inventory.yaml ansible_play.yaml`



AND Congratulation!!! you just completed your ansible with terraform project...


Ansible Success!
Not secure
3.90.247.190
Finish update
YAY!!!
We have learnt Ansible!