

Question 2 Part 2

VPC

CIDR : 10.0.0.0/16

Subnets:

forinvest-vpc-subnet-public1: 10.0.0.0/20

forinvest-vpc-subnet-public2: 10.0.16.0/20

Security Groups:

WebServerSecurityGroup

Inbound Rules:

TCP 80 from 0.0.0.0/0

TCP 443 from 0.0.0.0/0

TCP ALL from AnsibleSecurityGroup

Outbond Rules:

TCP ALL from 0.0.0.0/0

AnsibleSecurityGroup

Inbound Rules:

TCP 22 from My IP

TCP 22 from WebServerSecurityGroup

Outbond Rules:

TCP ALL from 0.0.0.0/0

EC2 Instances:

web-server01

subnet: forinvest-vpc-subnet-public1

Instance type: t2.micro

AMI: Debian 11

Elastic IP: 34.236.88.26

web-server02

subnet: forinvest-vpc-subnet-public2

Instance type: t2.micro

AMI: Debian 11

Elastic IP: 35.174.161.121

AnsibleInstance

subnet: forinvest-vpc-subnet-public1

Instance type: t2.micro

AMI: Debian 11

Setup Overview

Webservers deployed on different Public subnet with Elastic IP's. Ansible Instance deployed on one of the public subnet and restricted inbound access from only my local computers IP for ssh connection. -In production environment load balancer with SSL offloading can be used if both server serving for same application-

```
[WARNING]: Platform linux on host web-server02 is using the discovered Python
interpreter at /usr/bin/python3, but future installation of another Python
interpreter could change this. See https://docs.ansible.com/ansible/2.9/referen
ce_appendices/interpreter_discovery.html for more information.
web-server02 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host web-server01 is using the discovered Python
interpreter at /usr/bin/python3, but future installation of another Python
interpreter could change this. See https://docs.ansible.com/ansible/2.9/referen
ce_appendices/interpreter_discovery.html for more information.
web-server01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

Question 2 Part 2

The dynamic inventory is configured via the aws_ec2 plugin to query AWS for EC2 instances within the us-east-1 region. It filters instances by the tag Role: Webserver, ensuring that only the instances designated as web servers are targeted for the NGINX deployment.

```
admin@ip-10-0-27-66:~/ansible-project$ ansible-playbook deploy_nginx.yml --private-key=~/.ssh/forinvet-kp.pem -u admin

PLAY [Deploy Nginx on EC2 instances] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 10.0.17.19 is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
[ok: [10.0.17.19]
[WARNING]: Platform linux on host 10.0.15.86 is using the discovered Python interpreter at /usr/bin/python3, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
[ok: [10.0.15.86]

TASK [nginx : Install pip if not] *****
changed: [10.0.17.19]
changed: [10.0.15.86]

TASK [nginx : Install Docker SDK for Python] *****
changed: [10.0.17.19]
changed: [10.0.15.86]

TASK [nginx : Install Docker] *****
ok: [10.0.17.19]
ok: [10.0.15.86]

TASK [nginx : Pull the Nginx Docker image] *****
changed: [10.0.17.19]
changed: [10.0.15.86]

TASK [nginx : Run Nginx container] *****
changed: [10.0.17.19]
changed: [10.0.15.86]

PLAY RECAP *****
10.0.15.86      : ok=6  changed=4  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
10.0.17.19     : ok=6  changed=4  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```