

< ANSIBLE FOR AWS VPC – PART ONE >
IMPLEMENTATION PLAN

VERSION 1.0 | 09/06/2022 – 09/06/2022

MERAUHASSAN

Purpose

Cloud Automation with Ansible.

Project Scenario

- AWS Cloud Management Team
- Deploy / Setup Infra on Cloud
- Usage of Secure & HA VPC for new projects
- Regular requests for deployment

Problem Statement

- VPC consist of many moving parts {Subnets, Internet GW, Route Tables, NACL, Sec Grp}
- Bastian Host
- Human error can lead to non-functional or exposed VPC
- Managing manually is a time consuming task

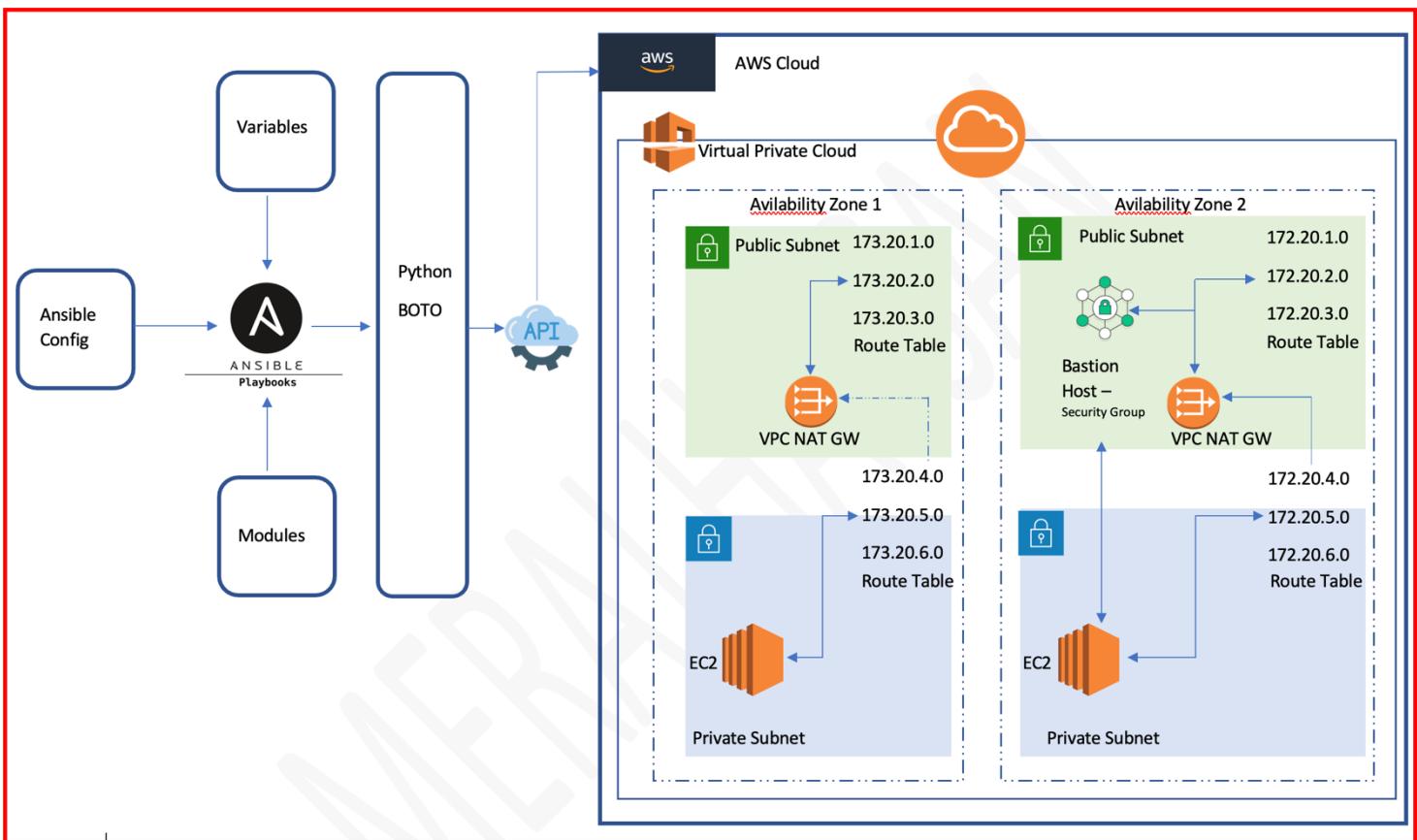
Solution

- Configuration management of VPC
- Automatic Setup [No Human Errors]
- Centralized Change Management
- Version Control [IAAC]

Tools

- Jenkins – Continuous Integration Server
- Ansible – Configuration management of AWS VPC
- AWS – VPC setup with Bastion host

Architecture Continuous Integration Pipeline



Flow of Execution

- *Login to AWS Account*
- *Create EC2 instance to run ansible playbook*
- *Install Ansible*
- *Instal BOTO*
- *Setup EC2 Role for Ansible*
- *Create the project directory*
- *Sample Cloud task (With key pair)*
- *Create variables file for VPC & Bastion host*
- *Create VPC Setup Playbook*
- *Create Bastion setup playbook*
- *Site.yaml playbook to call both playbook at once*

Prerequisite

1. *AWS Account*
2. *GITHUB Account*
3. *IntelliJ*

1. Ansible Setup for AWS

Create EC2 (Ubuntu 20) and install Ansible.

The screenshot shows the AWS Management Console interface for the 'Instances' section. At the top, there's a search bar and a user info bar. Below it, a table lists one instance:

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|-----------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|
| Control-Machine | i-08ee44367b00b10f3 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-2c |

```
[ubuntu@ip-172-31-43-193:~$ ansible --version
ansible 2.10.8
  config file = None
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.10.4 (main, Apr 2 2022, 09:04:19) [GCC 11.2.0]
```

IAM > Roles > Create role > **Ansible-Admin**

EC2 > Instances > Control-Machine > Modify IAM role > **Ansible-Admin**

```
$ apt install awscli
$ aws sts get-caller-identity
```

```
root@ip-172-31-43-193:~# aws sts get-caller-identity
{
  "UserId": "AROASL2YOSFBN2RZMW3Q0:i-08ee44367b00b10f3",
  "Account": "162857390402",
  "Arn": "arn:aws:sts::162857390402:assumed-role/Ansible-Admin/i-08ee44367b00b10f3"
}
```

Note: Do not put access key & secret key in playbook.

2. Sample AWS Cloud Playbook

Google Search – Ansible Module Index > Cloud Modules > EC2 Key
Check the requirement, based on that install the dependencies.

```
$ mkdir vpc-stack-eereeda
$ cd vpc-stack-eereeda/
$ sudo apt search boto
$ sudo apt install python3-boto3 -y
$ vim test-aws.yml
$ ansible-playbook test-aws.yaml
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda]$ cat test-aws.yaml
- hosts: localhost
  connection: local
  gather_facts: False
  tasks:
    - name: Sample ec2 key
      ec2_key:
        name: sample_keypair
        region: us-east-2
```

| Key pairs (1/2) Info | | | |
|---------------------------------------|------------------|------|---------------------------|
| <input type="text"/> Filter key pairs | | | |
| | Name | Type | Created |
| <input type="checkbox"/> | Ansible-Ohio-Key | rsa | 2022/06/09 13:18 GMT+5:30 |
| <input checked="" type="checkbox"/> | sample_keypair | rsa | 2022/06/09 14:30 GMT+5:30 |

To store the Private key in file(**sample_key.pem**) use **debug** and **copy** module in ansible playbook script

```
ok: [localhost] => {
  "keyout": {
    "changed": true,
    "failed": false,
    "key": {
      "fingerprint": "dc:67:e4:66:ca:2a:b4:a3:43:8e:d1:06",
      "name": "sample_keypair",
      "private_key": "-----BEGIN RSA PRIVATE KEY-----\nMIIE
```

```
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda$ cat test-aws.yaml
- hosts: localhost
  connection: local
  gather_facts: False
  tasks:
    - name: Sample ec2 key
      ec2_key:
        name: sample_keypair
        region: us-east-2
      register: keyout

    - debug:
        var: keyout

    - name: Store login key
      copy:
        content: "{{keyout.key.private_key}}"
        dest: ./sample_key.pem
```

```
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda$ ls
sample_key.pem  test-aws.yaml
```

If the key is already created then only save otherwise do not save

```
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda$ ansible-playbook test-aws.yaml
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] ****
TASK [Sample ec2 key] ****
ok: [localhost]
TASK [debug] ****
ok: [localhost] => {
  "keyout": {
    "changed": false,
    "failed": false,
    "key": {
      "fingerprint": "dc:67:e4:66:ca:2a:b4:a3:43:8e:d1:06:06:9f:60:3f:a0:49:27:63",
      "name": "sample_keypair"
    },
    "msg": "key pair already exists"
  }
}

TASK [Store login key] ****
fatal: [localhost]: FAILED! => {"msg": "The task includes an option with an undefined variable. The error was: 'dict object' has no attribute 'private_key'\n\nThe error appears to be in '/home/ubuntu/vpc-stack-eereeda/test-aws.yaml': line 14, column 7, but may\nbe elsewhere in the file depending on the exact synt\nax problem.\n\nThe offending line appears to be:\n\n      - name: Store login key\n          ^ here\n"}
```

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

```
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda]$ cat test-aws.yaml
- hosts: localhost
  connection: local
  gather_facts: False
  tasks:
    - name: Sample ec2 key
      ec2_key:
        name: sample_keypair
        region: us-east-2
      register: keyout

    - debug:
        var: keyout

    - name: Store login key
      copy:
        content: "{{keyout.key.private_key}}"
        dest: ./sample_key.pem
      when: keyout.changed
```

```
[ubuntu@ip-172-31-43-193:~/vpc-stack-eereeda]$ ansible-playbook test-aws.yaml
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] ****
TASK [Sample ec2 key] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
  "keyout": {
    "changed": false,
    "failed": false,
    "key": {
      "fingerprint": "dc:67:e4:66:ca:2a:b4:a3:43:8e:d1:06:06:9f:60:3f:a0:49:27:63",
      "name": "sample_keypair"
    },
    "msg": "key pair already exists"
  }
}

TASK [Store login key] ****
skipping: [localhost]

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

3. Variables for VPC

VPC Setup

```
vpc_name: "Eereeda-vpc"

# VPC Range
vpcCidr: '172.20.0.0/16'

# Subnet Range
PubSub1Cidr: 172.20.1.0/24
PubSub2Cidr: 172.20.2.0/24
PubSub3Cidr: 172.20.3.0/24
PrivSub1Cidr: 172.20.4.0/24
PrivSub2Cidr: 172.20.5.0/24
PrivSub3Cidr: 172.20.6.0/24

#
# Region Name
region: "us-east-2"

# Zone Names
zone1: us-east-2a
zone2: us-east-2b
zone3: us-east-2c

state: present
```

Bastion Setup

```
bastion_ami: ami-0fa49cc9dc8d62c84
region: us-east-2
MYIP: [REDACTED]
```

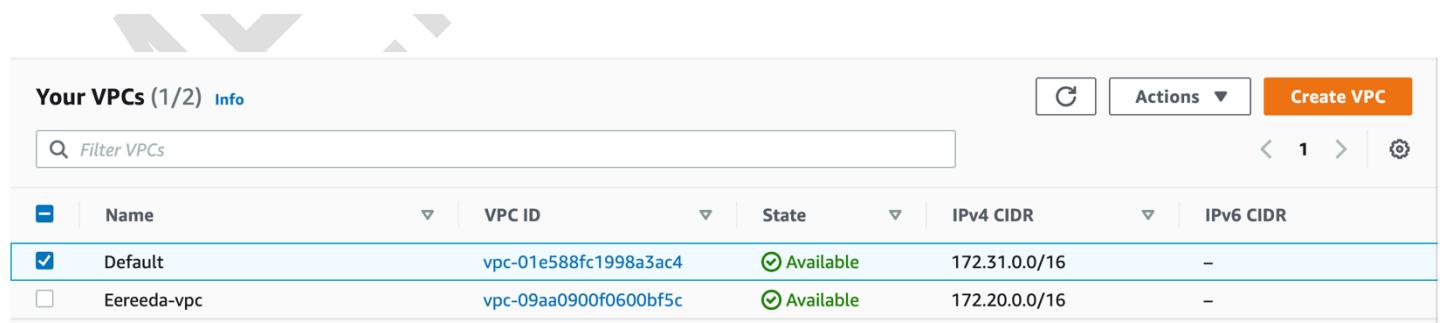
4. VPC Playbook

```
- hosts: localhost
connection: local
gather_facts: False
tasks:
  - name: Import VPC Variables
    include_vars: vars/vpc_setup

  - name: Create eereeda VPC
    ec2_vpc_net:
      name: "{{vpc_name}}"
      cidr_block: "{{vpcCidr}}"
      region: "{{region}}"
      dns_support: yes
      dns_hostnames: yes
      tenancy: default
      state: "{{state}}"
    register: vpcout
```

```
$ git clone https://github.com/merajafnan/DevOps_Projects.git
$ cd DevOps_Projects/
$ git checkout ansible-aws-vpc
$ git pull
$ ansible-playbook vpc-setup.yml
```

TASK [Import VPC Variables] ****
ok: [localhost]



| Your VPCs (1/2) Info | | | | | |
|---|-----------------------|------------------------|---------------|-----------|-------------------------|
| Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR | |
| <input checked="" type="checkbox"/> Default | vpc-01e588fc1998a3ac4 | Available | 172.31.0.0/16 | - | Actions |
| <input type="checkbox"/> Eereeda-vpc | vpc-09aa0900f0600bf5c | Available | 172.20.0.0/16 | - | Actions |

5. Subnets Playbook

```
TASK [debug] ****
ok: [localhost] => {
    "vpcout": {
        "changed": false,
        "debugging": {
            "expected_cidrs": [
                "172.20.0.0/16"
            ],
            "to_add": [],
            "to_remove": []
        },
        "failed": false,
        "vpc": {
            "cidr_block": "172.20.0.0/16",
            "cidr_block_association_set": [
                {
                    "association_id": "vpc-cidr-assoc-05f1f11aae56080d5",
                    "cidr_block": "172.20.0.0/16",
                    "cidr_block_state": {
                        "state": "associated"
                    }
                }
            ],
            "classic_link_enabled": false,
            "dhcp_options_id": "dopt-06caca59c636e0537",
            "id": "vpc-09aa0900f0600bf5c",
            "instance_tenancy": "default",
            "is_default": false,
            "owner_id": "162857390402",
            "state": "available",
            "tags": {
                "Name": "Eereeda-vpc"
            }
        }
    }
}
```

vpc-setup.yaml (Ansible Playbook)

```
- hosts: localhost
  connection: local
  gather_facts: False
  tasks:
    - name: Import VPC Variables
      include_vars: vars/vpc_setup

    - name: Create eereeda VPC
      ec2_vpc_net:
        name: "{{vpc_name}}"
        cidr_block: "{{vpcCidr}}"
        region: "{{region}}"
        dns_support: yes
        dns_hostnames: yes
        tenancy: default
        state: "{{state}}"
      register: vpcout
#
#       var: vpcout

    - name: Create Public Subnet 1 in Zone 1
```

```
ec2_vpc_subnet:
  vpc_id: "{{vpcout.vpc.id}}"
  region: "{{region}}"
  az: "{{zone1}}"
  state: "{{state}}"
  cidr: "{{PubSub1Cidr}}"
  map_public: yes
  resource_tags:
    Name: Eereeda-pubsub1
  register: pubsub1_out
#
# - debug:
#     var: pubsub1_out

- name: Create Public Subnet 2 in Zone 2
  ec2_vpc_subnet:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    az: "{{zone2}}"
    state: "{{state}}"
    cidr: "{{PubSub2Cidr}}"
    map_public: yes
    resource_tags:
      Name: Eereeda-pubsub2
  register: pubsub2_out

- name: Create Public Subnet 3 in Zone 3
  ec2_vpc_subnet:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    az: "{{zone3}}"
    state: "{{state}}"
    cidr: "{{PubSub3Cidr}}"
    map_public: yes
    resource_tags:
      Name: Eereeda-pubsub3
  register: pubsub3_out

- name: Create Private Subnet 1 in Zone 1
  ec2_vpc_subnet:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    az: "{{zone1}}"
    state: "{{state}}"
    cidr: "{{PrivSub1Cidr}}"
    map_public: yes
    resource_tags:
      Name: Eereeda-privsub1
  register: privsub1_out
```

```

- name: Create Private Subnet 2 in Zone 2
  ec2_vpc_subnet:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    az: "{{zone2}}"
    state: "{{state}}"
    cidr: "{{PrivSub2Cidr}}"
    map_public: yes
    resource_tags:
      Name: Eereeda-privsub2
  register: privsub2_out

- name: Create Private Subnet 3 in Zone 3
  ec2_vpc_subnet:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    az: "{{zone3}}"
    state: "{{state}}"
    cidr: "{{PrivSub3Cidr}}"
    map_public: yes
    resource_tags:
      Name: Eereeda-privsub3
  register: privsub3_out

```

Search for services, features, blogs, docs, and more [Option+S]

Ohio | meraj @ 1628-5739-04

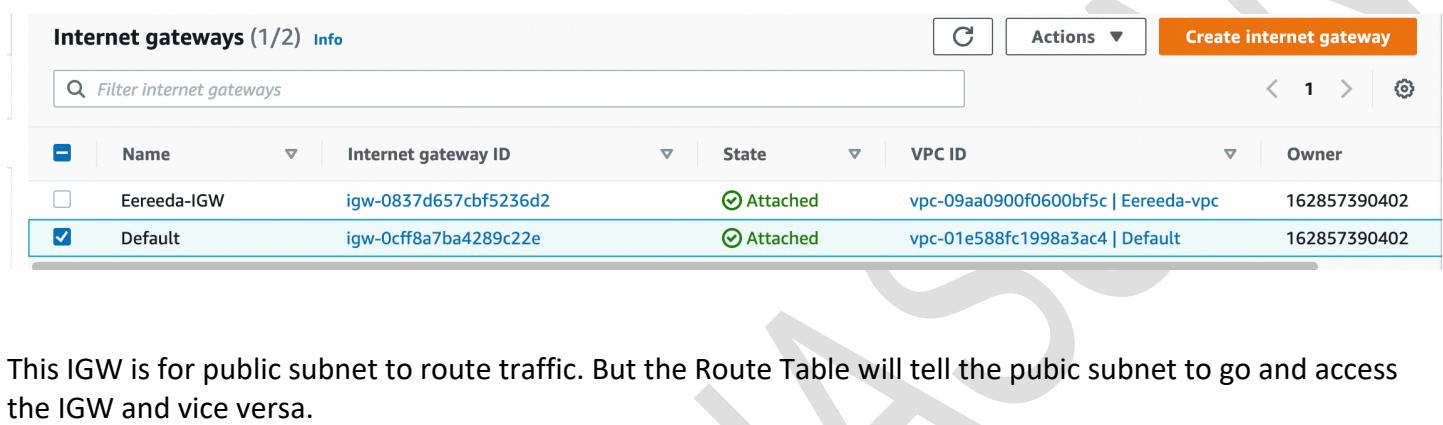
Subnets (9) Info

Actions ▾ Create subnet

| | Name | Sub... | State | VPC | IPv4 CIDR | IPv6... | Avai... | Availabil... | Availability Zone |
|--------------------------|------------------|-----------|----------|-----------|---------------|---------|---------|--------------|-------------------|
| <input type="checkbox"/> | Eereeda-pubsub3 | subnet... | ✓ Ava... | vpc-09... | 172.20.3.0/24 | - | 251 | us-east-2c | use2-az3 |
| <input type="checkbox"/> | Eereeda-pubsub2 | subnet... | ✓ Ava... | vpc-09... | 172.20.2.0/24 | - | 251 | us-east-2b | use2-az2 |
| <input type="checkbox"/> | Eereeda-pubsub1 | subnet... | ✓ Ava... | vpc-09... | 172.20.1.0/24 | - | 251 | us-east-2a | use2-az1 |
| <input type="checkbox"/> | Eereeda-privsub3 | subnet... | ✓ Ava... | vpc-09... | 172.20.6.0/24 | - | 251 | us-east-2c | use2-az3 |
| <input type="checkbox"/> | Eereeda-privsub2 | subnet... | ✓ Ava... | vpc-09... | 172.20.5.0/24 | - | 251 | us-east-2b | use2-az2 |
| <input type="checkbox"/> | Eereeda-privsub1 | subnet... | ✓ Ava... | vpc-09... | 172.20.4.0/24 | - | 251 | us-east-2a | use2-az1 |

6. Internet Gateway & Route Tables

```
- name: Internet Gateway Setup
  ec2_vpc_igw:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    state: "{{state}}"
    resource_tags:
      Name: Eereeda-IGW
  register: igw_out
```



| Internet gateways (1/2) Info | | | | | |
|--|-------------|-----------------------|----------|-------------------------------------|--------------|
| | Name | Internet gateway ID | State | VPC ID | Owner |
| <input type="checkbox"/> | Eereeda-IGW | igw-0837d657cbf5236d2 | Attached | vpc-09aa0900f0600bf5c Eereeda-vpc | 162857390402 |
| <input checked="" type="checkbox"/> | Default | igw-0cff8a7ba4289c22e | Attached | vpc-01e588fc1998a3ac4 Default | 162857390402 |

This IGW is for public subnet to route traffic. But the Route Table will tell the public subnet to go and access the IGW and vice versa.

```
- name: Setup Public Subnet Route Table
  ec2_vpc_route_table:
    vpc_id: "{{vpcout.vpc.id}}"
    region: "{{region}}"
    tags:
      Name: Eereeda-PubRouteTable
    subnets:
      - "{{pubsub1_out.subnet.id}}"
      - "{{pubsub2_out.subnet.id}}"
      - "{{pubsub3_out.subnet.id}}"
    routes:
      - dest: 0.0.0.0/0
        gateway_id: "{{igw_out.gateway_id}}"
  register: pubRT out
```

| Route tables (1/3) Info | | | | | | |
|---|-----------------------|-----------------------|------------------------------|-------------------|------|--------------------|
| | Name | Route table ID | Explicit subnet associations | Edge associations | Main | VPC |
| <input checked="" type="checkbox"/> | Eereeda-PubRouteTable | rtb-0282cf46bd9cbec16 | 3 subnets | - | No | vpc-09aa0900f0600b |
| <input type="checkbox"/> | - | rtb-0788f3e66d1230c38 | - | - | Yes | vpc-09aa0900f0600b |

rtb-0282cf46bd9cbec16 / Eereeda-PubRouteTable

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Explicit subnet associations (3)

[Edit subnet associations](#)

| Subnet ID | IPv4 CIDR | IPv6 CIDR |
|--|---------------|-----------|
| subnet-0cc5528dd8f5169d3 / Eereeda-pubsub1 | 172.20.1.0/24 | - |
| subnet-07382d3d9e5898b2b / Eereeda-pubsub3 | 172.20.3.0/24 | - |
| subnet-0227ba80468d601b7 / Eereeda-pubsub2 | 172.20.2.0/24 | - |

[Details](#) [Routes](#) [Subnet associations](#) [Edge associations](#) [Route propagation](#) [Tags](#)

Routes (2)

[Edit routes](#)

| Destination | Target | Status | Propagated |
|---------------|-----------------------|--------|------------|
| 172.20.0.0/16 | local | Active | No |
| 0.0.0.0/0 | igw-0837d657cbf5236d2 | Active | No |

MEPRX

7. NAT Gateway & Route Table

```
- name: Create new NAT GW and allocate new Elastic IP if a NAT GW does not yet exist in the VPC
  ec2_vpc_nat_gateway:
    state: "{{state}}"
    subnet_id: "{{pubsub1_out.subnet.id}}"
    wait: yes
    region: "{{region}}"
    if_exist_do_not_create: true
  register: NATGW_out

- name: Setup Private Subnet Route Table
  ec2_vpc_route_table:
    vpc_id: "{{vcfout.vpc.id}}"
    region: "{{region}}"
    tags:
      Name: Eereeda-PrivRouteTable
    subnets:
      - "{{privsub1_out.subnet.id}}"
      - "{{privsub2_out.subnet.id}}"
      - "{{privsub3_out.subnet.id}}"
    routes:
      - dest: 0.0.0.0/0
        gateway_id: "{{NATGW_out.nat_gateway_id}}"
  register: privRT_out
```

| NAT gateways (1/1) Info | | | | | |
|--|-----------------------|----------------|---------------------------|---------------|--------------------|
| Actions Create NAT gateway | | | | | |
| <input type="text"/> Filter NAT gateways | | | | | |
| Name | NAT gateway ID | Connectivit... | State | State message | Elastic IP address |
| Eereeda_NAT_GW | nat-043242ebd3c8a506a | Public | Available | - | 3.15.54.117 |

| Elastic IP addresses (1/1) | | | | | |
|---|-----------------------|-----------|----------------------------|------------|--|
| Actions Allocate Elastic IP address | | | | | |
| <input type="text"/> Filter Elastic IP addresses | | | | | |
| Name | Allocated IPv4 add... | Type | Allocation ID | Reverse DN | |
| Eereeda_Elastic_IP | 3.15.54.117 | Public IP | eipalloc-0cb75c51e13b8604d | - | |

| Route tables (1/4) Info | | | | | | |
|--|------------------------|-----------------------|-----------------------------|-------------------|------|----------------|
| Actions Create route table | | | | | | |
| <input type="text"/> Filter route tables | | | | | | |
| - | Name | Route table ID | Explicit subnet associat... | Edge associations | Main | VPC |
| <input checked="" type="checkbox"/> | Eereeda-PrivRouteTable | rtb-0ebabad0e399cd13a | 3 subnets | - | No | vpc-09aa0900fc |
| <input type="checkbox"/> | Eereeda-PubRouteTable | rtb-0282cf46bd9cbe16 | 3 subnets | - | No | vpc-09aa0900fc |

8. Store ID's & Variable File

```
- debug:
  var: "{{item}}"
loop:
  - vpcout.vpc.id
  - pubsub1_out.subnet.id
  - pubsub2_out.subnet.id
  - pubsub3_out.subnet.id
  - privsub1_out.subnet.id
  - privsub2_out.subnet.id
  - privsub3_out.subnet.id
  - igw_out.gateway_id
  - pubRT_out.route_table.id
  - NATGW_out.nat_gateway_id
  - privRT_out.route_table.id
- set_fact:
  vpcid: "{{vpcout.vpc.id}}"
  pubsub1id: "{{pubsub1_out.subnet.id}}"
  pubsub2id: "{{pubsub2_out.subnet.id}}"
  pubsub3id: "{{pubsub3_out.subnet.id}}"
  privsub1id: "{{privsub1_out.subnet.id}}"
  privsub2id: "{{privsub2_out.subnet.id}}"
  privsub3id: "{{privsub3_out.subnet.id}}"
  igwid: "{{igw_out.gateway_id}}"
  pubRTid: "{{pubRT_out.route_table.id}}"
  NATGWid: "{{NATGW_out.nat_gateway_id}}"
  privRTid: "{{privRT_out.route_table.id}}"
  cacheable: yes

- name: Copy variables in files from VPC output
  copy:
    content: "vpcid: {{vpcout.vpc.id}}\npubsub1id: {{pubsub1_out.subnet.id}}\npubsub2id: {{pubsub2_out.subnet.id}}\npubsub3id: {{pubsub3_out.subnet.id}}\nprivsub1id: {{privsub1_out.subnet.id}}\nprivsub2id: {{privsub2_out.subnet.id}}\nprivsub3id: {{privsub3_out.subnet.id}}\nigwid: {{igw_out.gateway_id}}\nprivRTid: {{pubRT_out.route_table.id}}\nNATGWid: {{NATGW_out.nat_gateway_id}}\nprivRTid: {{privRT_out.route_table.id}}"
    dest: vars/output_vars
```

ubuntu@ip-172-31-43-194:/DevOps_Projects\$ cat vars/output_vars

```
vpcid: vpc-09aa...vv-13-5c
pubsub1id: subnet-0cc5...8d19-5167c
pubsub2id: subnet-0227...47-770-7
pubsub3id: subnet-073f...24-89-32
privsub1id: subnet-0e34...6-9b1-50-ff
privsub2id: subnet-0821...b-784-39-75f
privsub3id: subnet-027a...f-067c6ec9
igwid: igw-0837...0-1057-13602
pubRTid: rtb-0282cf...d-49-ec16
NATGWid: nat-043242e...8-506a
```

9. Bastion Host Setup

Bastion-instance.yaml (Bastion Playbook)

```
---
- name: Setup Eereeda Bastion Host
  hosts: localhost
  connection: local
  gather_facts: False
  tasks:
    - name: Import VPC Variables
      include_vars: vars/bastion_setup

    - name: Import VPC setup Variables
      include_vars: vars/output_vars

    - name: Create Eereeda EC2 Key
      ec2_key:
        name: Eereeda-key-Ansible
        region: "{{region}}"
      register: key_out

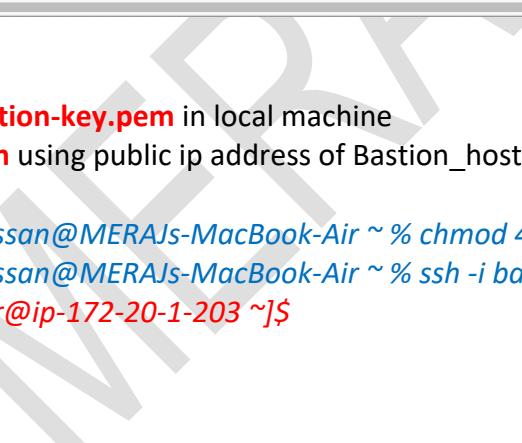
    - name: Save private Key into file bastian-key.pem
      copy:
        content: "{{key_out.key.private_key}}"
        dest: "./bastion-key.pem"
        mode: 0600
      when: key_out.changed

    - name: Create security group for bastion host
      ec2_group:
        name: Bastion-host-sg
        description: Allow port 22 from everywhere and all
ports withn SG
        region: "{{region}}"
        vpc_id: "{{vpcid}}"
        rules:
          - proto: tcp
            from_port: 22
            to_port: 22
            cidr_ip: "{{MYIP}}"
      register: BastionSG_out
```

```

- name: Creating Bastion Host
  ec2:
    key_name: Eereeda-key-Ansible
    region: "{{region}}"
    instance_type: t2.micro
    image: "{{bastion_ami}}"
    wait: yes
    wait_timeout: 300
    instance_tags:
      Name: "Bastion_host"
      Project: Eereeda
      Owner: Meraj
    exact_count: 1
    count_tag:
      Name: "Bastion_host"
      Project: Eereeda
      Owner: Meraj
    group_id: "{{BastionSG_out.group_id}}"
    vpc_subnet_id: "{{pubsublid}}"
  register: bastionHost_out

```



Search for services, features, blogs, docs, and more [Option+S]

Ohio | meraj @ 1628-5739-0402

Instances (2) Info

| <input type="checkbox"/> | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|--------------------------|-----------------|---------------------|----------------------|---------------|--------------------------------|--------------|-------------------|
| <input type="checkbox"/> | Bastion_host | i-0c10680921e1960ed | Running | t2.micro | Initializing | No alarms | us-east-2a |
| <input type="checkbox"/> | Control-Machine | i-08ee44367b00b10f3 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-2c |

Actions ▾ Launch instances ▾

Save **bastion-key.pem** in local machine

Try to **ssh** using public ip address of Bastion_host

```

merajhassan@MERAJs-MacBook-Air ~ % chmod 400 bastion-key.pem
merajhassan@MERAJs-MacBook-Air ~ % ssh -i bastion-key.pem ec2-user@18.118.20.61
[ec2-user@ip-172-20-1-203 ~]$

```

1. References

The following table summarizes the documents referenced in this plan.

| DOCUMENT NAME | INSTRUCTOR | LOCATION |
|------------------------------|------------|---|
| DevOps Beginners to Advanced | Imran Teli | https://www.udemy.com/course/decodingdevops/learn/lecture/28273912?start=0#overview |
| | | |
| | | |