ASSIGNMENT FOR Coding INTERVIEW

Cloud Engineering

Mahmoud Ali

Table of Contents

1 Solution Explanation

This solution aims to develop a platform to host Django Application. It is hosted inside AWS and is leveraging on AWS services (Elastic Beanstalk, VPC, Loadbalancers, Route53, ACM, etc…) as shown in the figure below.

The Application is split into two different environments:

- PRD
  - Production environment needed to serve the service from internet
- TST
  - Testing environment to allow all possible testing.

The TST environment is a full replica of production to help the developers to customize and release new versions of the application without any downtime.

Elastic Beanstalk is customized to provide:

- Two Environments
  - PRD → Production Environment
  - TST → Testing Environment
- Customized Security Group
- Application Load balancer (ALB) with sticky sessions enabled
- HTTPS access
- AutoScaling based on traffic load
- Route 53 DNS

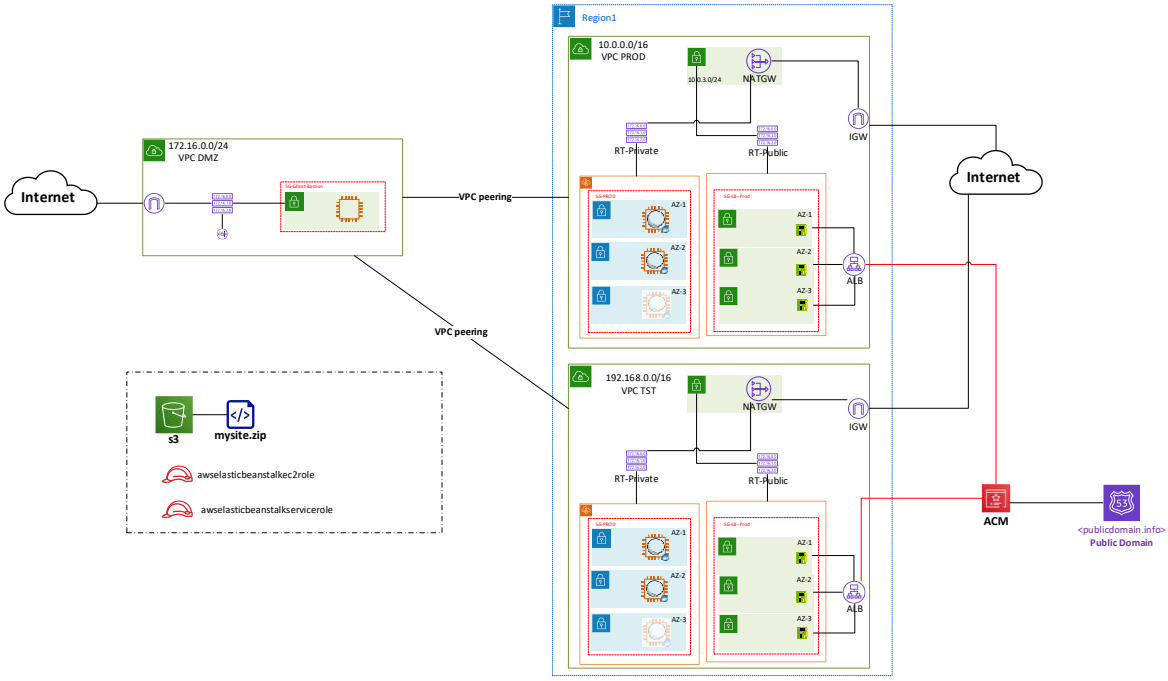
NAT Gateway are used to provide only outbound internet connectivity of the Application EC2 instances.

Moreover, the environments are also connected to a DMZ using VPC peering to keep a single point of access and limit access to the application network. So, the SSH to the production and testing EC2 instances deployed by Elastic Beanstalk is only allowed from a Bastion instance that is deployed in DMZ VPC.

There are 3 security groups used here:

- SG-ALB → to permit http, https traffic from everywhere.
- SG-EC2 → to permit http traffic only from SG-ALB and SSH only from SG-DMZ
- SG-DMZ → to permit SSH only from specific External IP address.

The source code of the application is stored on a S3 bucket.



2 Logical Topology

This solution is customized to provide the following logical topology and cloud components:

- 3 VPCs
  - 1 VPC PRD
  - 1 VPC TST
  - 1 VPC DMZ
- VPCs PRD, TST are peering with DMZ VPC to provide access connectivity from outside.
- Security Groups are in place and are configured to reference to each other with specific source and protocols/ports.
- Routing table Public for each VPC (PRD, TST) to provide internet connectivity to the Application Load balancer (ALB) and NAT Gateway.
- Routing table Public for DMZ that has internet connectivity
- Routing table Private for Django App
  - Enable traffic to NAT GW
  - Django App is only reachable by ALB and Bastion Host
- 1 NAT Gateway for each VPC (PRD, TST) to provide internet connectivity for Django App.
- AWS Certificate Manager to generate and store the SSL certificate
  - These certificates are used in all environments (PRD, TST)
- Route 53 to provide a DNS records.
- S3 bucket to host the first version of the Django application
- Elastic Beanstalk with the following options settings:
  - Load Balanced application
  - HTTPS
  - Sticky session enabled
  - Autoscaling
    - Min 2
    - Max 4

3 Standard Deployment & Infra HLD

The infrastructure deployment leverages Infrastructure as Code (Terraform). The script already provides some parameters by default and others are to be filled:

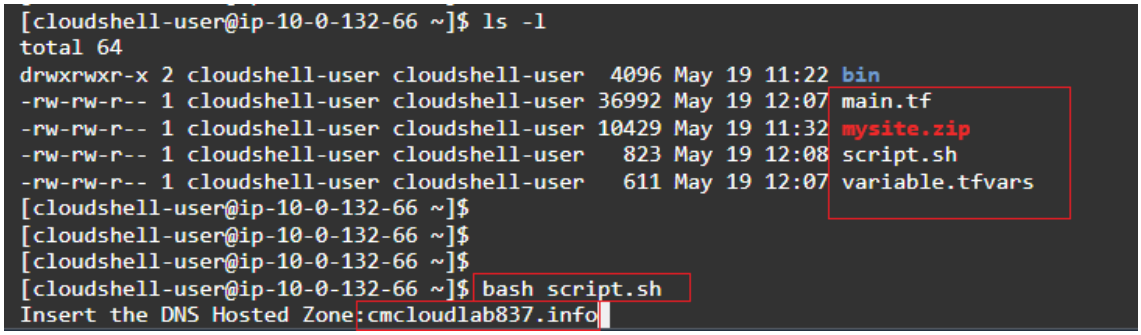
Note: The script assumes a DNS hosted zone with domain name is already created.

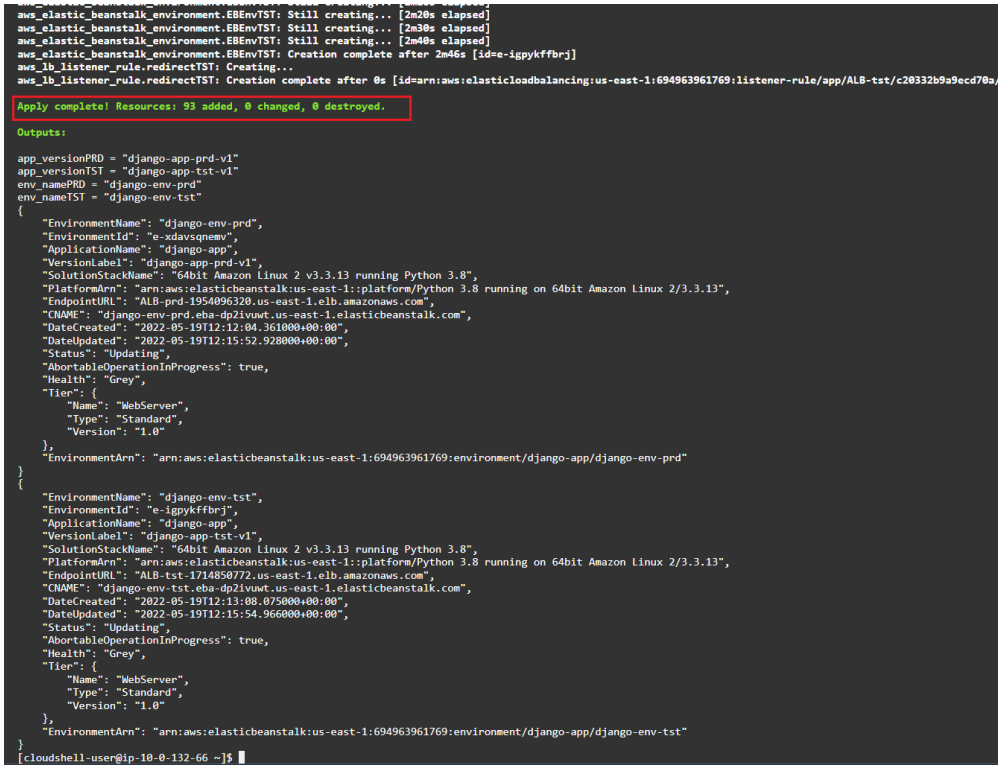
Here below are the parameters to be inserted:

Name	Description	Default
Route53 domain name	Insert name of Route 53 DNS hosted zone	to be filled
ExternalIP	Insert your Public IP of your host	to be filled

To run the script, open the AWS Cloudshell or access it remotely.

1. Upload or clone the repository from GitHub
   - main.tf
   - variable.tfvars
   - script.sh
   - mysite.zip
2. Run the script.sh → **bash script.sh**
3. Insert the DNS domain



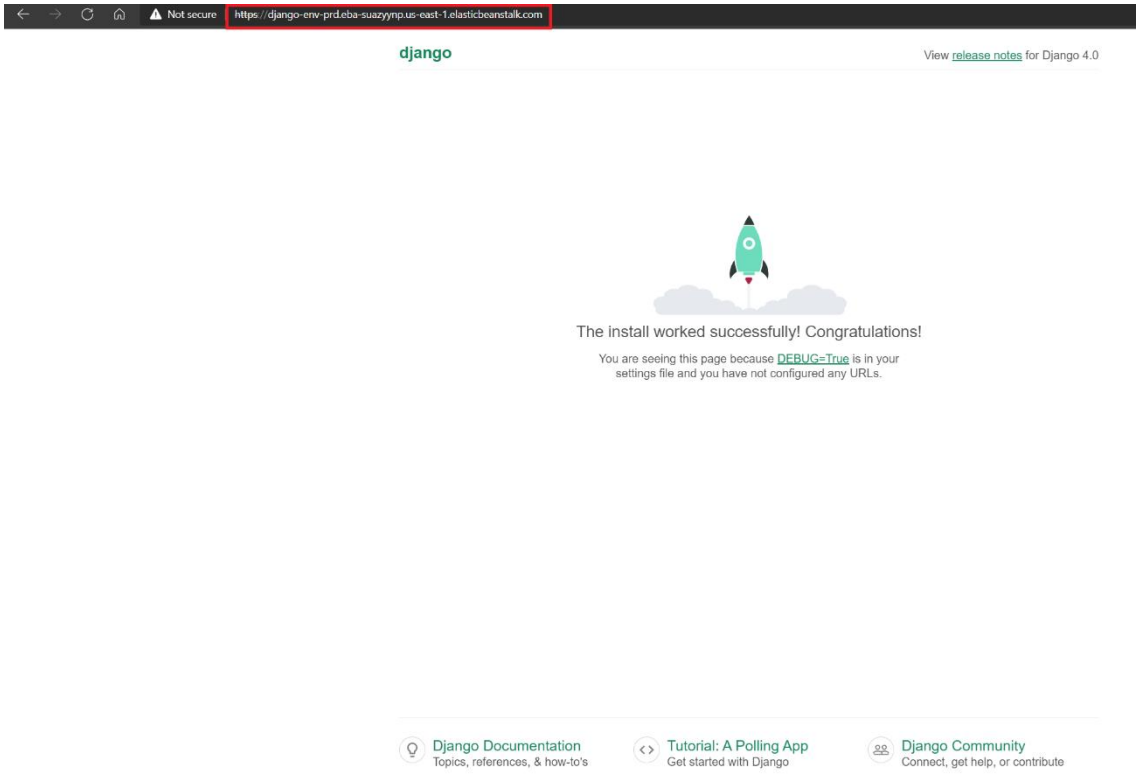


The script is deploying all the infrastructure needed for elastic beanstalk to run the Django Application inside AWS. In the specific it is deploying:

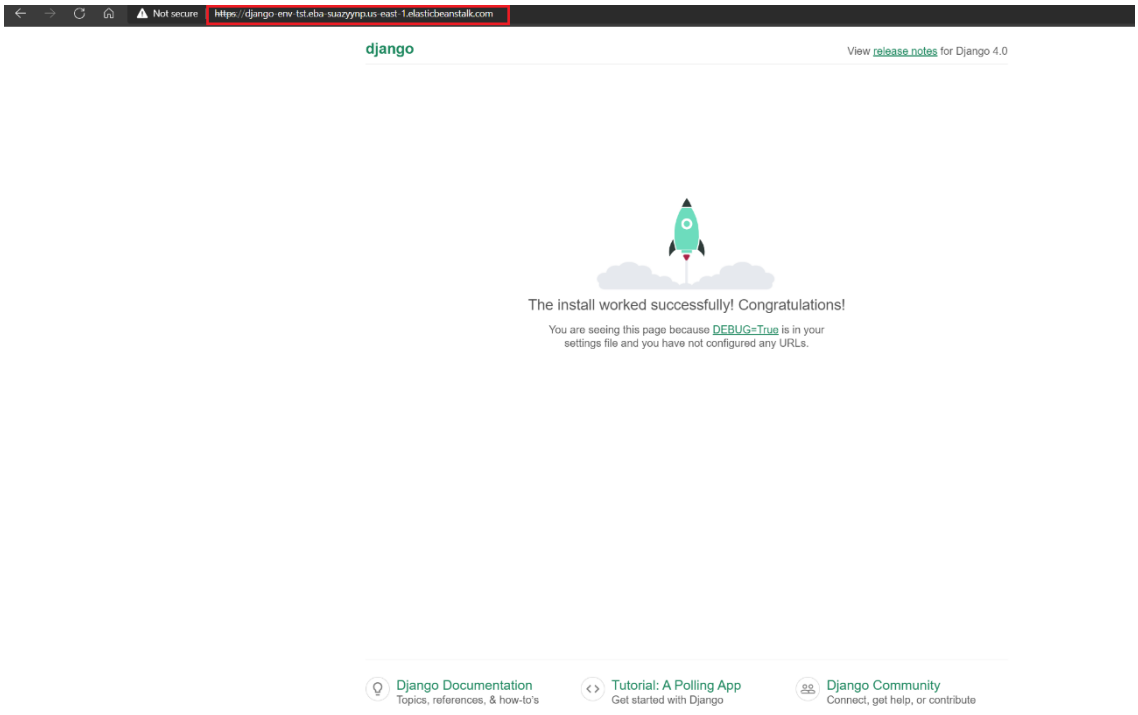
- VPC (PRD, TST, DMZ)
- Subnets
- Internet GW
- NAT GW
- Route Tables
- ACM
- Route 53 DNS records
- S3 Bucket
- Bastion EC2 Instance
- VPC Peering
- Security Groups
- Access Lists
- Application Load Balancer
- Elastic Beanstalk
- IAM Roles
- EC2 Key Pair

4 Output

Access the production environment



Access the testing environment



5 Reference Documents

Drawing.pdf