Terraform Interview Test: Basic AWS Resource Creation

Instructions

Please write Terraform code to accomplish the tasks described in each question. Ensure that your code follows best practices and is well-documented. You may use any Terraform features or modules you deem necessary.

Question 1: Create an S3 Bucket

Create an AWS S3 bucket with the following specifications:

- The bucket should have versioning enabled.
- The bucket should have server-side encryption enabled using AWS-managed keys (SSE-S3).
- The bucket should have a lifecycle rule to transition objects to Glacier storage after 30 days.

Question 2: Create an EC2 Instance Module

Create a Terraform module for creating an AWS EC2 instance with the following specifications:

- The module should use the latest Amazon Linux 2 AMI.
- The instance type should be configurable (default to `t2.micro`).
- The module should create a new VPC with a single public subnet.
- The module should create a security group that allows SSH access (port 22) from anywhere.
- Follow Terraform best practices for module creation, including input variables, output values, and documentation.

Additionally, write a README.md file for the module that includes:

- A description of the module.
- Instructions on how to use the module.
- Examples of how to call the module from a root module.

Question 3: Create an RDS Instance

Create an AWS RDS instance with the following specifications:

- The database engine should be MySQL.
- The instance class should be `db.t3.micro`.
- The database should be created in a new subnet group with at least two subnets in different availability zones.
- The database should have automated backups enabled with a retention period of 7 days.

Question 4: Create an IAM Role and Attach a Policy

Create an AWS IAM role with the following specifications:

- The role should be assumable by EC2 instances.
- Attach a policy to the role that allows full access to S3 and read-only access to EC2.
- Ensure the policy is written in JSON format within the Terraform code.

Question 5: Create a VPC with Public and Private Subnets

Create an AWS VPC with the following specifications:

- The VPC should have a CIDR block of `10.0.0.0/16`.
- Create one public subnet and one private subnet within the VPC.
- The public subnet should have a route to the internet via an internet gateway.
- The private subnet should have a route to the internet via a NAT gateway.

Submission

1. Write the Terraform code for each question (Except the EC2 Module) in separate `.tf` files in a Single Folder or as a single `.tf` file with clear comments indicating the start and end of each question's solution.
2. Create Separate folder and put all TF files for EC2 Module into the folder and zip it with other resources folder before send it
3. Test your Terraform code to ensure it works as expected. You can use `terraform fmt` to format your code and `terraform validate` to validate it.
4. Zip the `.tf` files and name the zip file with your name (e.g., <Your_Name>`_Terraform_Test.zip`).
5. Send the zip file to the HR team for further evaluation.

Good luck!