

CSYE 6225: Network Structure & Cloud Computing

Assignment #6

Prerequisites:

- Recommended to use the AWS Academy Lab account.
- If using a personal AWS account, ensure to:
 - Only create micro instances.
 - Terminate all resources after finishing the assignment.
- For any resource that requires a name, use your name as a prefix.
- Use Terraform to provision all the resources.

Assignment Requirements

1. IAM Role and Instance Profile Setup (Skip if using AWS Academy)

i. Create an IAM Role:

- Use the **AmazonS3ReadOnlyAccess** policy to allow the EC2 instance to read from the S3 bucket. `policy_arn = "arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess"`

ii. Create an Instance Profile:

- Attach the IAM role to an instance profile.
- This instance profile will be linked to the EC2 instance during provisioning.

2. EC2 Instance Setup

i. Launch an EC2 instance using the following configuration:

- The latest Ubuntu AMI (fetched dynamically using a data source).
- Instance type: t2.micro.
- Enable a public IP address to make the instance accessible over the internet.
- Attach the instance profile created earlier (or the "LabInstanceProfile" if using AWS Academy).
- Assign a tag with the name of your instance: "Name": "web-server-instance".

- ii. Use a user-data script to install and start Nginx during instance initialization:
 - You need to create a simple index.html file that contains your name and place it in the Nginx serving directory (/var/www/html/). This will ensure that when you access the public IP of the EC2 instance in a browser, it displays your custom HTML page.
- iii. Output the public IP address of the instance after the infrastructure is created.

3. Security Group Configuration

- i. Create a Security Group in the VPC to:
 - Allow inbound HTTP (port 80) from anywhere (0.0.0.0/0) (ingress rule).

References for Official Documentation

- [AWS Provider Configuration](#)
- [Security Group Resource](#)
- [AMI Data Source](#)
- [EC2 Instance Resource](#)

Submission Instructions

- Upload the following Terraform files on Canvas:
 - main.tf
 - variables.tf
 - outputs.tf
 - terraform.tfvars
- A PDF report containing at least:
 - A screenshot of terminal output showing the public IP address of the EC2 instance ([Screenshot #1]).
 - A screenshot from the AWS Management Console showing the security group configuration ([Screenshot #2]).
 - A screenshot showing the custom index.html page served on the public IP of your EC2 instance via Nginx in the web browser ([Screenshot #3]).
 - A screenshot of the terminal output from terraform destroy, confirming successful deletion of all resources. ([Screenshot #4]).

Helpful Terraform Commands

1. **Initialize Terraform:** terraform init
2. **Plan the Infrastructure:** terraform plan
3. **Apply the Plan:** terraform apply
4. **Verify Nginx:** Open the public IP address in a web browser or run: curl http://<ec2_public_ip>
5. **Destroy Resources (Cleanup):** terraform destroy

Important Notes

1. Terraform Setup:
 - a. Create an IAM role with AdministratorAccess policy from your console if you are using AWS personal account. (Skip if using AWS Academy)
 - b. Create an EC2 Ubuntu instance and attach the above IAM role to the instance (or directly attach "LabInstanceProfile" if using AWS Academy).
 - c. SSH into the instance and install aws-cli: sudo snap install aws-cli --classic
 - d. Follow the [Terraform installation guide for Ubuntu](#).
 - e. Ensure Terraform is installed and configured with your AWS credentials before starting your assignment.
2. Resource Cleanup:
 - a. Remember to destroy the resources after completing the assignment to avoid unnecessary costs.
 - b. Run terraform destroy before terminating the EC2 instance you created.

Grading

- **No late assignments are accepted.**
- **PDF report:** 10 points for each required screenshot (4 screenshots - max 40 points).
- **Correct code** that successfully provisions resources (10 points for each resource listed above - total 40 points).
- **Max points:** 80 points (40 for report + 40 for code).

End of Assignment #6