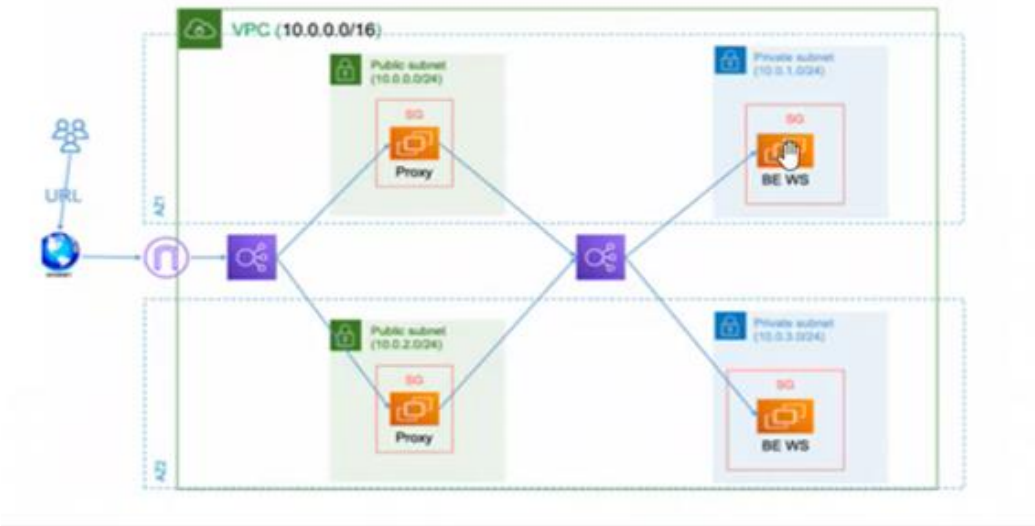
Project Overview

This Terraform project is designed to create an infrastructure with multiple EC2 instances and two load balancers, as shown in the provided diagram. The infrastructure consists of:

1. A Virtual Private Cloud (VPC) with public and private subnets.
2. EC2 instances configured as web servers in the private subnets.
3. A proxy server in the public subnets that routes traffic to the private web servers.
4. Two load balancers:
    - The first one is a public load balancer that forwards traffic to the proxy server.
    - The second one is a private load balancer that forwards traffic to the backend EC2 web servers.

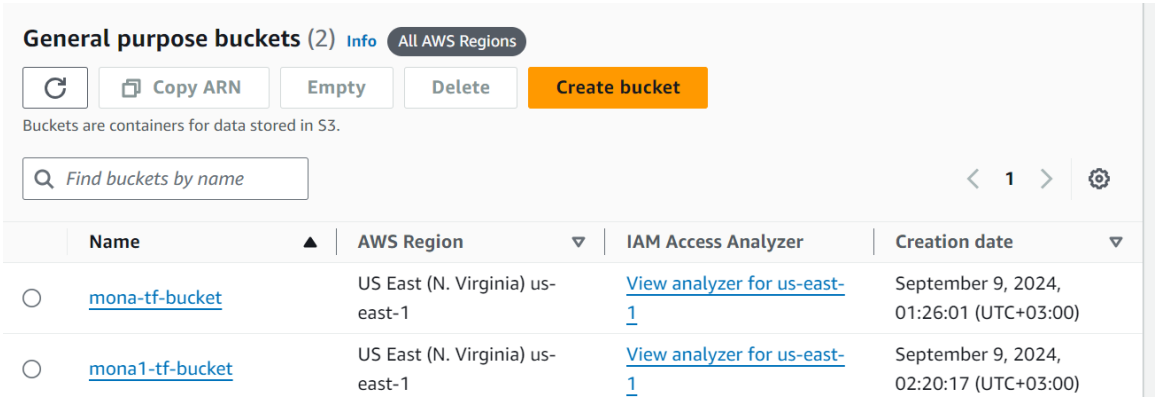
The project uses AWS services, including EC2, Elastic Load Balancers (ELB), and S3 for storing Terraform state files.



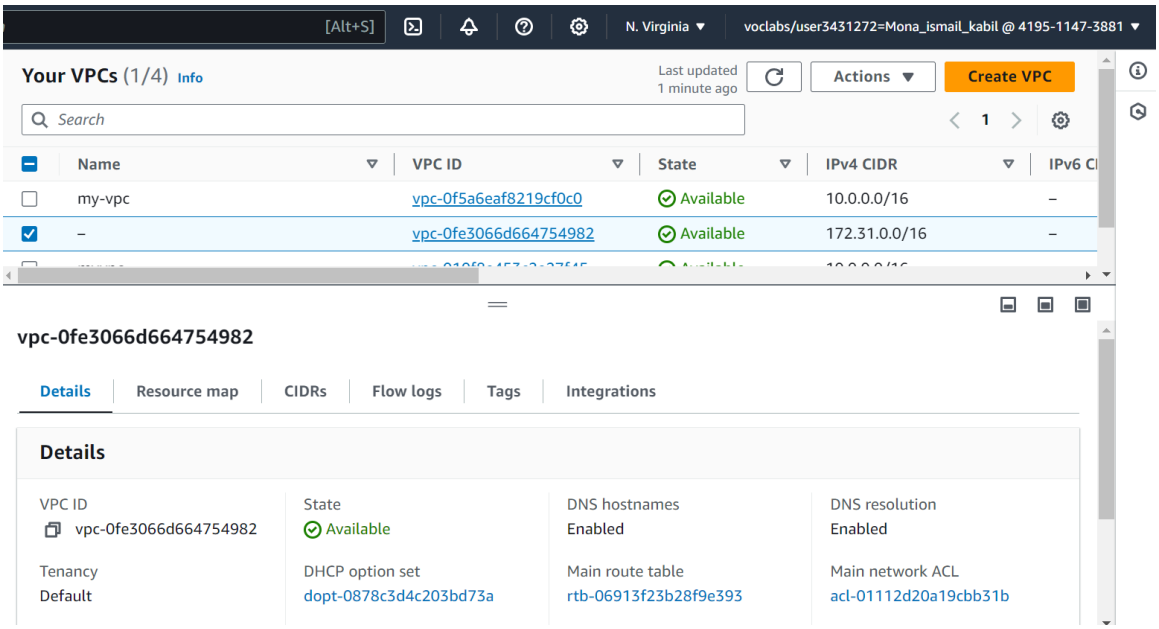
Prerequisites

1. **Terraform** installed on your local machine.
2. **AWS CLI** configured with appropriate credentials.
3. An existing **S3 bucket** and **DynamoDB table** for remote state management.
4. **SSH key pair** for EC2 access

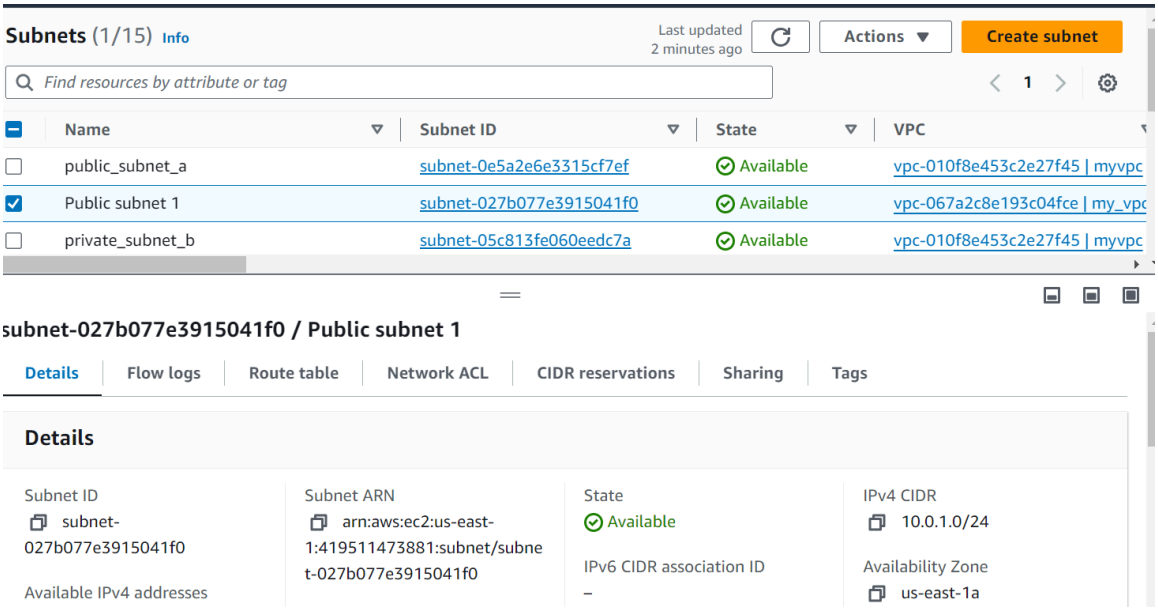
S3



1. **VPC**: Create a VPC using a custom VPC module.
    - creating a VPC module in `modules/VPC/main.tf`:



2. **Subnets: create** public and private subnets using a custom **Subnet** module
    - Define subnets, route tables, and gateways as needed in `modules/Subnet/main.tf`

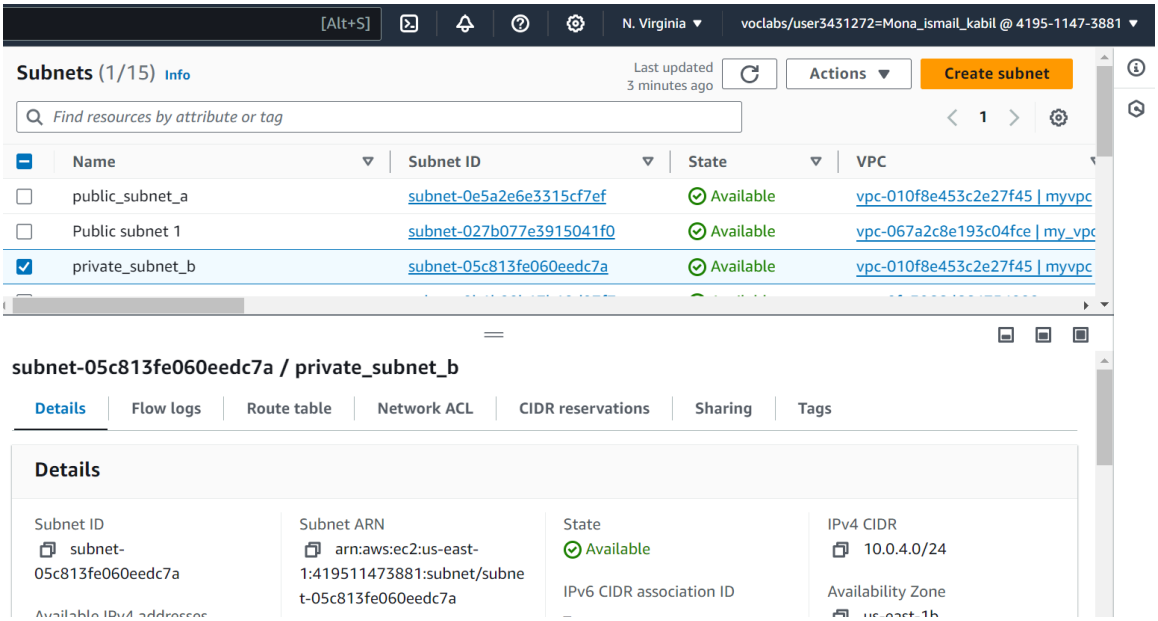


subnet-027b077e3915041f0 / Public subnet 1

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

Details

Subnet ID subnet-027b077e3915041f0	Subnet ARN arn:aws:ec2:us-east-1:419511473881:subnet/subnet-027b077e3915041f0	State Available	IPv4 CIDR 10.0.1.0/24
Available IPv4 addresses		IPv6 CIDR association ID –	Availability Zone us-east-1a



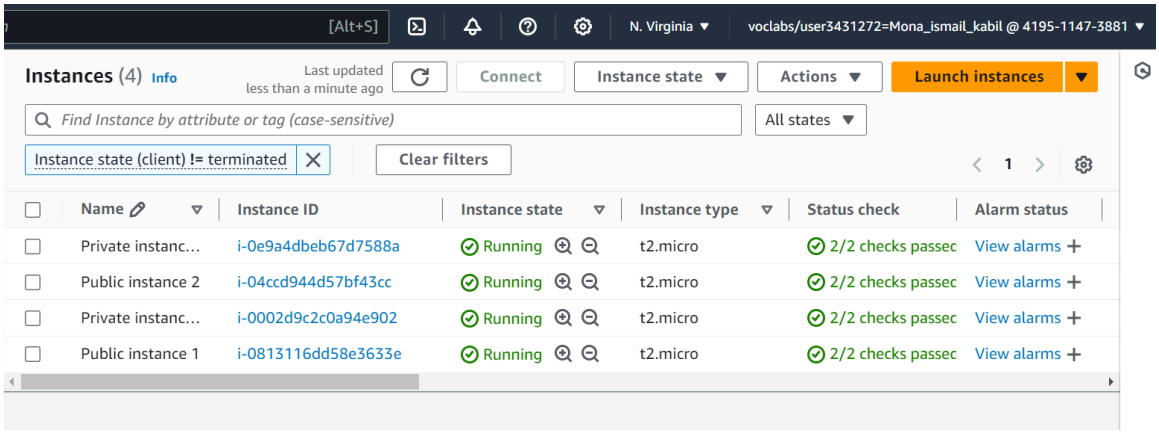
subnet-05c813fe060eedc7a / private_subnet_b

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

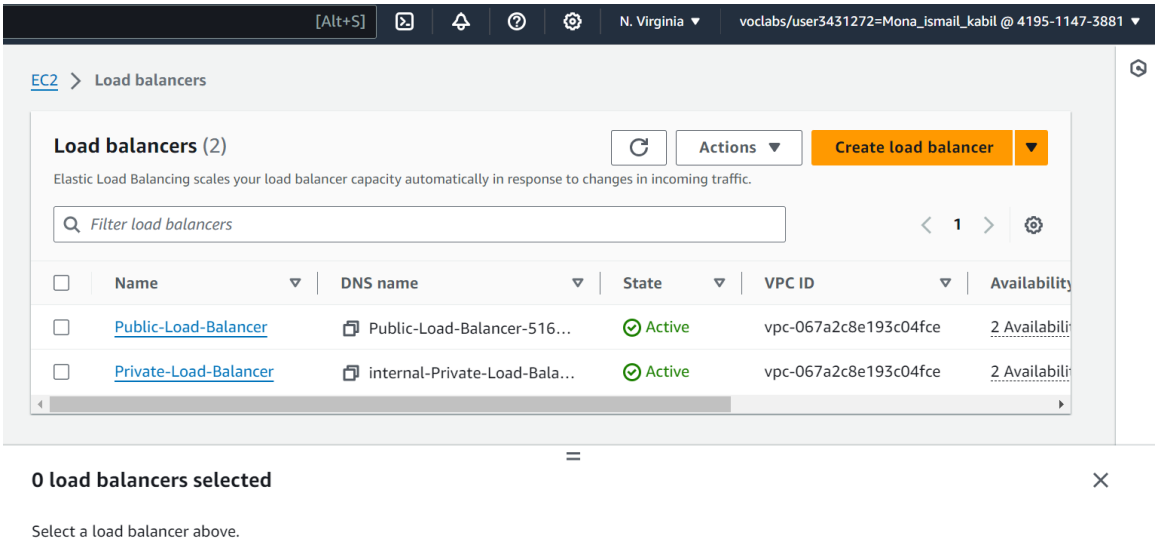
Details

Subnet ID subnet-05c813fe060eedc7a	Subnet ARN arn:aws:ec2:us-east-1:419511473881:subnet/subnet-05c813fe060eedc7a	State Available	IPv4 CIDR 10.0.4.0/24
		IPv6 CIDR association ID	Availability Zone

3. **EC2 Instances**:

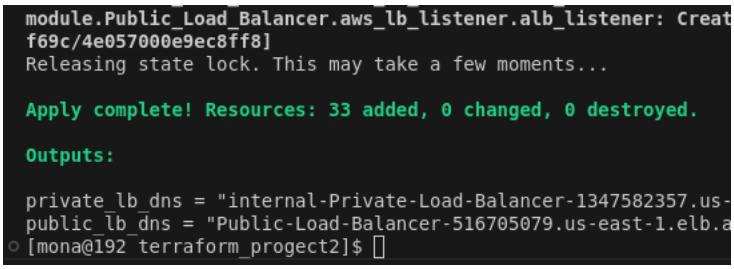


4. **Load Balancers**:

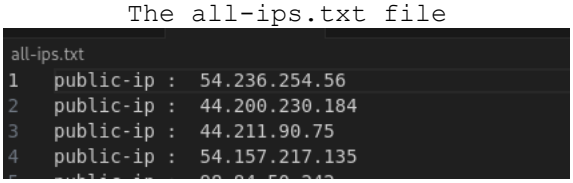


5. **Output Values**:
    - Define output values for the public IP addresses and DNS names of the load balancers.
    - Examples
        - **Public Load Balancer DNS**: The DNS name of the public load balancer.
        - **Private Load Balancer DNS**: The DNS name of the private load balancer.
        - **EC2 Public IPs**: The public IP addresses of EC2 instances created in the public subnets.

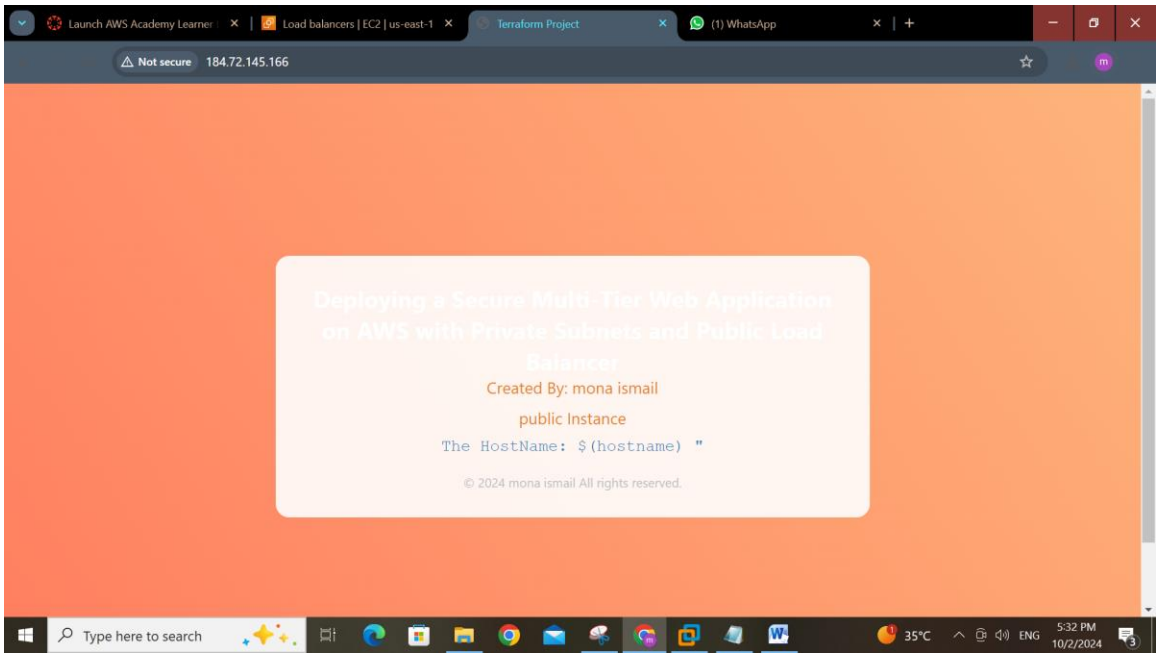
The final Result:



The all-ips.txt file



6. **Testing**:
    - Access the public DNS of the load balancer and verify it forwards traffic to the proxy.



Conclusion

This documentation outlines the steps to implement a complex AWS infrastructure using Terraform. The project is structured to separate concerns into modules, making it easier to manage and scale