National College of Ireland

Configuration Manual

MSc Research Project
Cloud Computing

Anusha Singamaneni
StudentID:23237066

School of Computing
National College of Ireland

Supervisor: Prof. Shaguna Gupta

National College of Ireland
Project Submission Sheet School of Computing

Student Name:	Anusha Singamaneni
Student ID:	23237066
Programme:	Cloud Computing
Year:	2024
Module:	MSc Research Project
Supervisor:	Prof. Shaguna Gupta
Submission Due Date:	12/12/2024
Project Title:	Configuration Manual
Word Count:	800
Page Count:	8

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	12th December 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Anusha Singamaneni
23237066

1 How to Install Terraform?

- Go the Terraform Downloads Page and download the Windows (64-bit) .zip file.
- Extract the downloaded .zip to a directory like C:\Users\sanus\Downloads\Terraform.
- Setting Open Environment Variables by using the Start Menu.
- However, when using command prompts and calling functions, you need to modify the Path under System Variables by adding the folder containing the extracted Terraform.
- To check the installation of Terraform launch the Command Prompt or PowerShell type terraform -version.
- Chocolatey is one of the best package managers for windows download it and then use choco install terraform to install Terraform. Check on the terminal using terraform version.

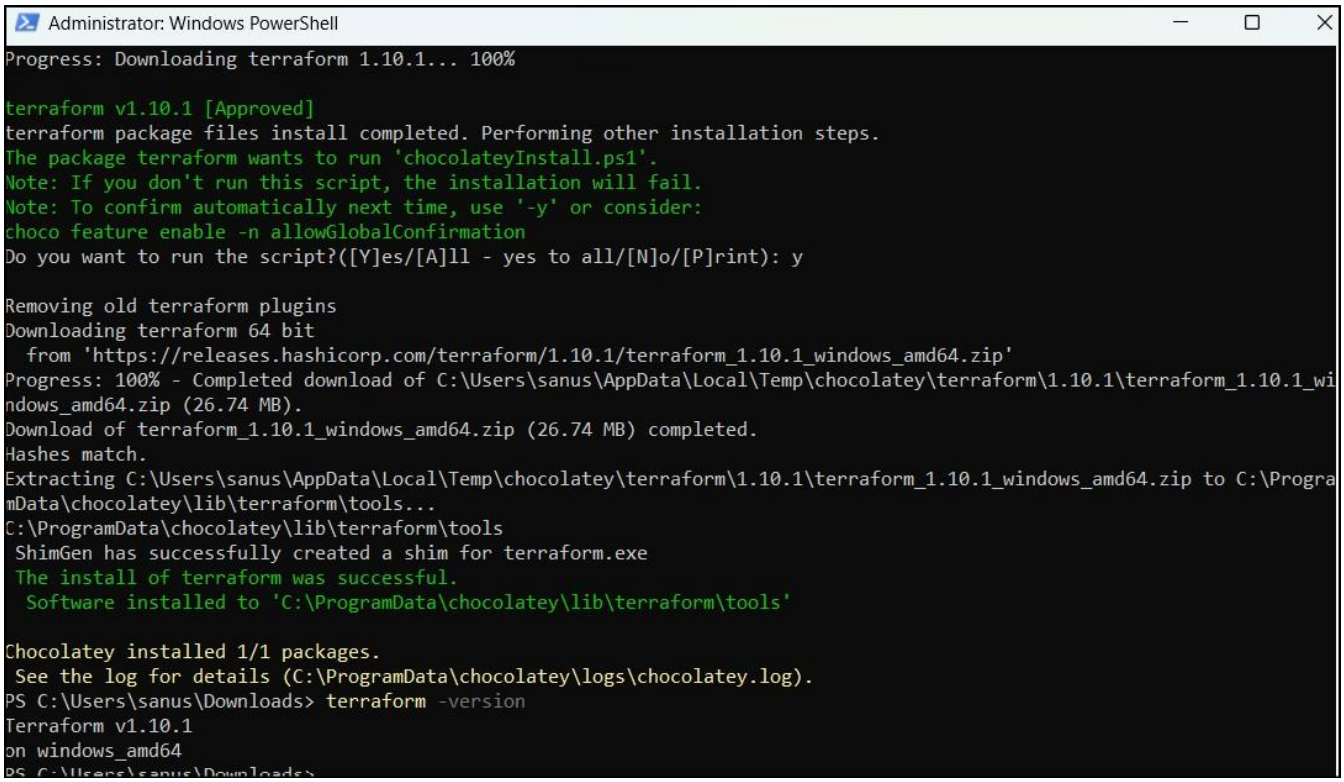

Figure 1: Terraform Installation

2 How to create IAM role with login credentials in AWS?

The access to the IAM services requires having an active AWS cloud account to complete below steps. It is better to take a different AWS IAM role which will be used for performing further interactions with the services.

Create a New IAM Role

- Access the AWS Management Console
- Search for **IAM** in the search bar and navigate to the IAM service. • Create a New IAM Role
- Select Trusted Entity
- Attach Policies

Eg:

JSON

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "s3:*",
   "Resource": "*"
  }
 ] }
```

- Set Role Name
- Create Role

Generate Login Credentials for an IAM User

- Create an IAM User
- Attach the Role or Policies
- Generate Access Keys

Terraform Workflow for IAM Role and User

- Write Terraform Config (main.tf)
- terraform init
- terraform apply
- AWS Resources Created (IAM Role, Policies, User, Credentials) • Retrieve & Use Credentials

IAMGroups.tf

Secure-Terraform-Infrastructure > IAMGroups.tf

```
# Existing Groups and Policies
resource "aws_iam_group" "admins" {
  name = "AdminGroup"
}

resource "aws_iam_group_policy_attachment" "admins_policy_attach" {
  group      = aws_iam_group.admins.name
  policy_arn = "arn:aws:iam::aws:policy/AdministratorAccess"
}

resource "aws_iam_group" "devops" {
  name = "DevOpsGroup"
}

resource "aws_iam_policy" "devops_policy" {
  name        = "DevOpsPolicy"
  description = "Policy for DevOps team to manage infrastructure"
  policy      = data.aws_iam_policy_document.devops_policy.json
}

data "aws_iam_policy_document" "devops_policy" {
```

Figure 2: IAM Groups in Terraform

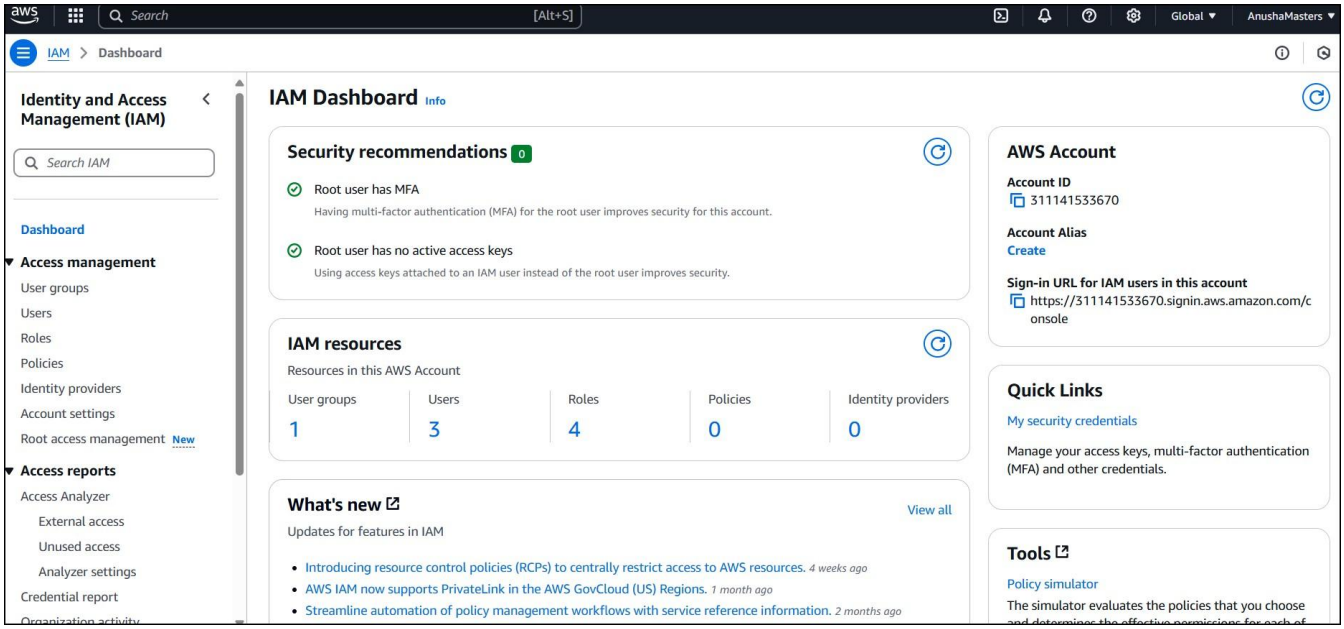

Figure 3: IAM Dashboard in AWS

3 How to set generated credentials in variable file. (tfvars>terraform.tfvars)?

Facing sensitive credentials including API keys, database passwords, cloud provider access keys, etc. it is vital to adhere to secure, automated and consistent approach to work. This guide builds on the different areas of how to create, set up and protect credential in Terraform Projects.

- Define credentials in variables as variables in the variables.tf file.
- Create/get your credential and store them in terraform.tfvars in a safe manner.
- Reading these variables in the Terraform configuration.

Figure 4: Terraform Variables

4 How to Run?

- **Install Prerequisites** ○ Terraform: Available for download on Terraform's official website: Install on your computer.
  - AWS CLI: It is installed and set up using Command Line Interface abbreviated as CLI by using the aws configure.
- **Configure Project** ○ Also, make sure any files as the awsKey.pem file is appropriately stored in the correct format and from the wrong hands. Update permissions for private keys: chmod 400 awsKey.pem
- **Initialize Terraform**
  - Set up the environment and download necessary plugins: terraform init
- **Validate and Preview Changes** ○ Validate the configuration syntax terraform validate
  - Preview changes Terraform will make terraform plan

- **Deploy Infrastructure** ○ Apply the configuration to deploy resources terraform apply -var-file=tfvars/terraform.tfvars
- **Review Outputs**
  - Display important outputs defined in outputs.tf: terraform output
- **Clean Up Resources** ○ If you need to tear down the infrastructure: terraform destroy -var-file=tfvars/terraform.tfvars

Secure-Terraform-Infrastructure > outputs.tf

```
output "vpc_id" {
  value = aws_vpc.main.id
}

output "public_subnets" {
  value = aws_subnet.public[*].id
}

output "private_subnets" {
  value = aws_subnet.private[*].id
}

output "nat_instance_id" {
  value = aws_instance.nat.id
}

output "bastion_host_id" {
  value = aws_instance.bastion.id
}

```

Figure 5: Terraform Output

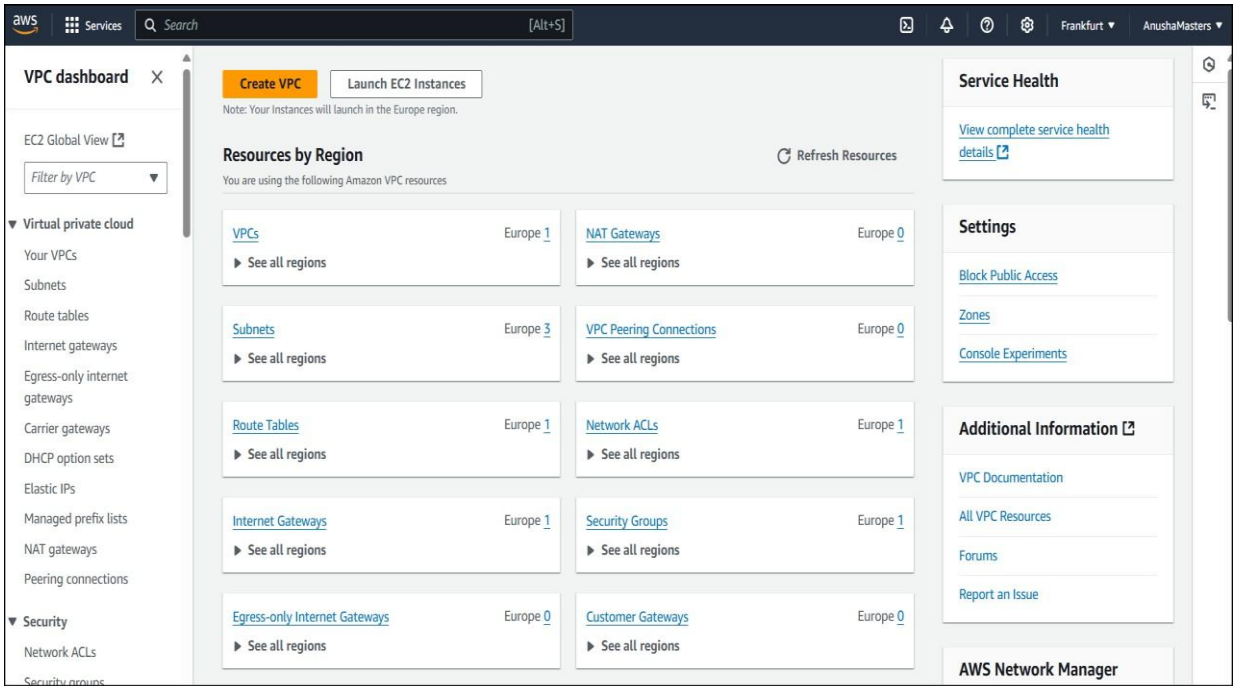

Figure 6: AWS Resources

References

1. Terraform official documentation for installation:

https://www.terraform.io/downloads.html

2. Chocolatey Terraform installation guide:

https://community.chocolatey.org/packages/terraform

3. AWS official documentation for creating IAM roles: https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html

4. AWS JSON policy syntax guide:

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies.html

5. Terraform variable configuration guide:

https://developer.hashicorp.com/terraform/language/values/variables

6. Terraform best practices for sensitive variables:

https://developer.hashicorp.com/terraform/language/values/variables#sensitive-values

7. AWS CLI installation guide:

https://docs.aws.amazon.com/cli/latest/userguide/installcliv2.html

8. Terraform CLI commands and initialization steps:

https://developer.hashicorp.com/terraform/cli/commands