TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

BÁO CÁO THỰC HÀNH 01

Dùng Terraform và CloudFormation để quản lý và triển khai hạ tầng AWS

Môn học: Công nghệ DevOps và ứng dụng

Lớp: NT548.P21

THÀNH VIÊN THỰC HIỆN (Nhóm 21):

STT	Họ và tên	MSSV
1	Lê Bình Nguyên	22520969
2	Đặng Hữu Phát	22521065
3	Châu Thế Vĩ	22521653

Điểm tự đánh giá
10/10

ĐÁNH GIÁ KHÁC:

Tổng thời gian thực hiện	7 ngày
Phân chia công việc	- Terraform: Phát, Vĩ - CloudFormation: Nguyên - Viết báo cáo: Phát, Vĩ, Nguyên
Ý kiến *(nếu có)* + Khó khăn + Đề xuất, kiến nghị	

Phần bên dưới của báo cáo này là báo cáo chi tiết của nhóm thực hiện

2

BÁO CÁO CHI TIẾT

MỤC TIÊU BÀI THỰC HÀNH

Bài thực hành nhằm mục đích:

1. Sử dụng Terraform và CloudFormation để triển khai tự động hạ tầng trên AWS.
2. Tạo và quản lý các tài nguyên AWS bao gồm VPC, Subnets, Internet Gateway, NAT Gateway, Route Tables, Security Groups, và EC2 Instances.
3. Đảm bảo tính bảo mật và khả năng kết nối giữa các tài nguyên theo yêu cầu.
4. Viết mã nguồn theo dạng module và kiểm tra kết quả triển khai bằng các test case.

TÀI NGUYÊN SỬ DỤNG

Tất cả mã nguồn, tệp cấu hình, và tài liệu liên quan được lưu trữ công khai tại:
GitHub Repository: tại đây

Nội dung kho lưu trữ bao gồm:

- **Mã nguồn**: Terraform (.tf), CloudFormation (.yaml).
- **Tệp kiểm tra**: Script test_infra.sh để xác minh các tài nguyên.
- **Tài liệu**: Báo cáo PDF, README.md hướng dẫn triển khai.
- **Hình ảnh minh họa**: Các ảnh chụp màn hình console AWS và kết quả triển khai.

LỜI CẢM ƠN

Nhóm chúng em xin bày tỏ lòng biết ơn sâu sắc đến thầy Lê Anh Tuấn – Giảng viên khoa Mạng Máy tính và Truyền thông, Trường Đại học Công nghệ Thông tin, Đại học Quốc gia Thành phố Hồ Chí Minh. Thầy đã tận tình giảng dạy và hướng dẫn nhóm trong môn giúp chúng em xây dựng nền tảng kiến thức vững chắc để thực hiện bài thực hành.

Dẫu đã cố gắng hết sức, nhưng trong quá trình thực hiện, khó có thể tránh khỏi những thiếu sót. Nhóm rất mong nhận được những ý kiến đóng góp quý báu từ thầy để có thể hoàn thiện hơn trong tương lai.

Một lần nữa, nhóm chúng em xin chân thành cảm ơn!

Nhóm thực hiện: Nhóm 21

I. Terraform (Xem chi tiết)

1. Cấu trúc thư mục

Dự án Terraform được tổ chức theo dạng module để đảm bảo tính tái sử dụng và dễ bảo trì. Dự án Terraform được tổ chức theo dạng module để đảm bảo tính tái sử dụng và dễ bảo trì. Cấu trúc thư mục chính bao gồm

```
Terraform/
├── main.tf
├── outputs.tf
├── variables.tf
├── modules/
│   ├── vpc/
│   ├── subnet/
│   ├── internet-gateway/
│   ├── nat-gateway/
│   ├── route-table-public/
│   ├── route-table-private/
│   ├── security-group-public/
│   ├── security-group-private/
│   └── ec2/
└── .terraform.lock.hcl
```

2. Triển khai từng dịch vụ

- VPC:

```
module "vpc" {
  source     = "./modules/vpc"
  cidr_block = var.vpc_cidr
  name       = "my-vpc"
}
```

- Subnet: Public Subnet (kết nối với Internet Gateway) và Private Subnet (sử dụng NAT Gateway để kết nối ra ngoài).

```
module "public_subnet" {
  source          = "./modules/subnet"
  vpc_id          = module.vpc.vpc_id
  cidr_block      = var.public_subnet_cidr
  az              = var.az_1
  map_public_ip   = true
  name            = "public-subnet"
}

module "private_subnet" {
  source          = "./modules/subnet"
  vpc_id          = module.vpc.vpc_id
  cidr_block      = var.private_subnet_cidr
  az              = var.az_1
  map_public_ip   = false
  name            = "private-subnet"
}
```

- Internet Gateway: Cho phép các tài nguyên trong Public Subnet kết nối với Internet.

```
module "internet_gateway" {
  source = "./modules/internet-gateway"
  vpc_id = module.vpc.vpc_id
  name   = "my-igw"
}
```

- NAT Gateway: Cho phép các tài nguyên trong Private Subnet có thể kết nối Internet.

```
module "nat_gateway" {
  source           = "./modules/nat-gateway"
  public_subnet_id = module.public_subnet.subnet_id
}
```

- Route Table: Định tuyến lưu lượng Internet.

```
module "route_table_public" {
  source              = "./modules/route-table-public"
  vpc_id              = module.vpc.vpc_id
  internet_gateway_id = module.internet_gateway.igw_id
  name                = "public-rt"
  public_subnet_id    = module.public_subnet.subnet_id
}

module "route_table_private" {
  source          = "./modules/route-table-private"
  vpc_id          = module.vpc.vpc_id
  nat_gateway_id  = module.nat_gateway.nat_gateway_id
  name            = "private-rt"
  private_subnet_id = module.private_subnet.subnet_id
}
```

- Security Groups: kiểm soát lưu lượng vào/ra của các EC2 instances.

```
module "sg_public" {
  source         = "./modules/security-group-public"
  vpc_id         = module.vpc.vpc_id
  allowed_ssh_ip = var.allowed_ssh_ip
}

module "sg_private" {
  source         = "./modules/security-group-private"
  vpc_id         = module.vpc.vpc_id
  public_sg_id   = module.sg_public.sg_id
}
```

- EC2 Instance: Nằm trong các Subnet tương ứng.

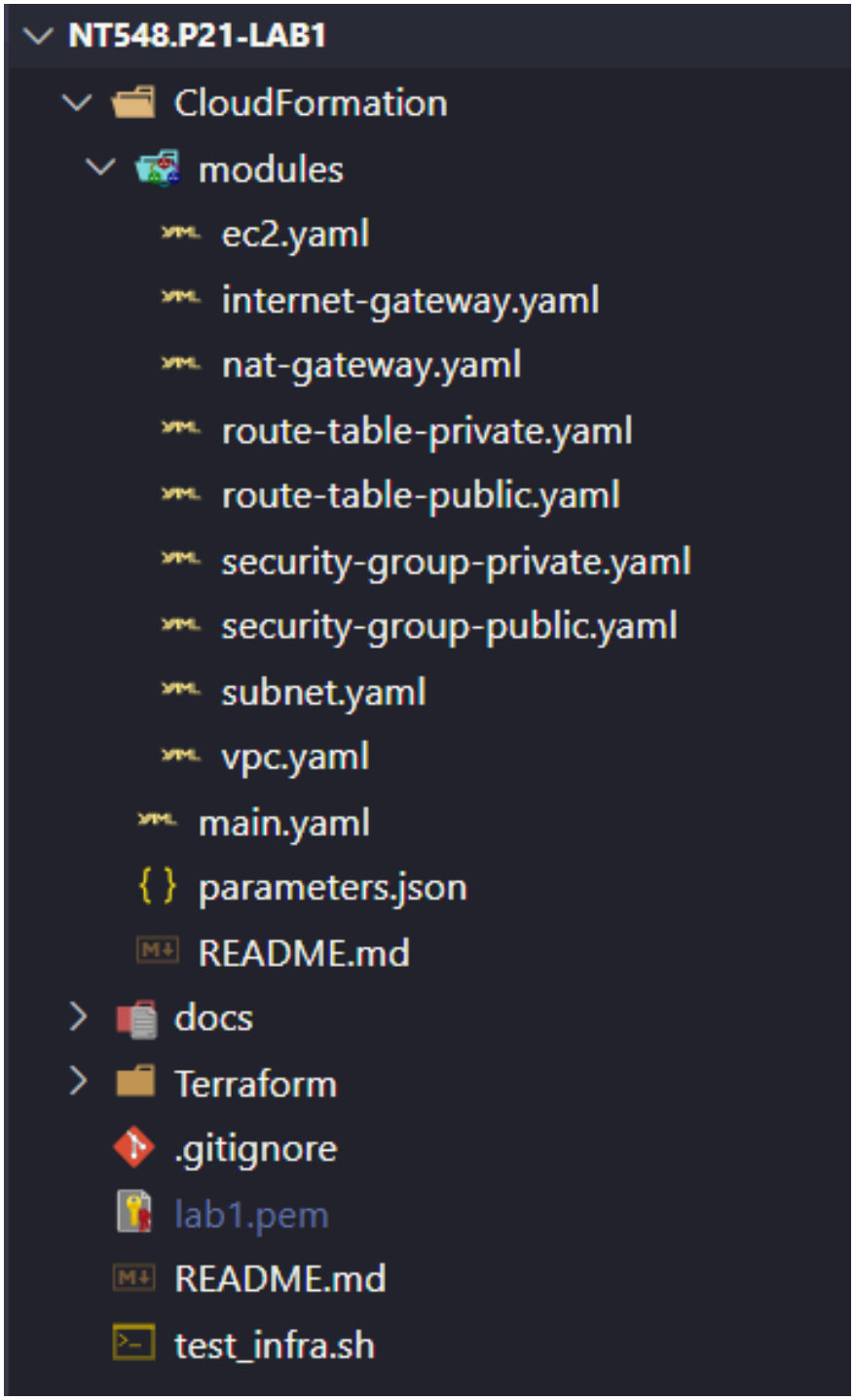
```
module "ec2_public" {
  source              = "./modules/ec2"
  ami                 = var.ami
  instance_type       = var.instance_type
  subnet_id           = module.public_subnet.subnet_id
  key_name            = var.key_name
  sg_id               = module.sg_public.sg_id
  associate_public_ip = true
  name                = "public-ec2"
}

module "ec2_private" {
  source              = "./modules/ec2"
  ami                 = var.ami
  instance_type       = var.instance_type
  subnet_id           = module.private_subnet.subnet_id
  key_name            = var.key_name
  sg_id               = module.sg_private.sg_id
  associate_public_ip = false
  name                = "private-ec2"
}
```

3. **Cấu hình tham số cho các dịch vụ**

```
region             = "ap-southeast-1"
vpc_cidr           = "10.0.0.0/16"
public_subnet_cidr = "10.0.1.0/24"
private_subnet_cidr = "10.0.2.0/24"
az_1               = "ap-southeast-1a"
ami                = "ami-xxxxxxxxxxxxxxxxx" # Replace with valid AMI ID
instance_type      = "t3.micro"
key_name           = "your-key-pair-name"
allowed_ssh_ip     = "YOUR.IP.ADDRESS/32"
```

II. Clouformation (Xem chi tiết)

1. Cấu trúc thư mục



2. Triển khai từng dịch vụ

- VPC:

```
VPCStack:
  Type: AWS::CloudFormation::Stack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/vpc.yaml"
    Parameters:
      VpcCIDR: !Ref VpcCidr
```

- Subnet: Public Subnet (kết nối với Internet Gateway) và Private Subnet (sử dụng NAT Gateway để kết nối ra ngoài).

```
PublicSubnet:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/subnet.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      SubnetCIDR: !Ref PublicSubnetCidr
      AvailabilityZone: !Ref AvailabilityZone
      MapPublicIP: "true"
      SubnetName: "public-subnet"

PrivateSubnet:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/subnet.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      SubnetCIDR: !Ref PrivateSubnetCidr
      AvailabilityZone: !Ref AvailabilityZone
      MapPublicIP: "false"
      SubnetName: "private-subnet"
```

- Internet Gateway: Cho phép các tài nguyên trong Public Subnet kết nối với Internet.

```
InternetGatewayStack:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/internet-gateway.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      Name: "my-igw"
```

- NAT Gateway: Cho phép các tài nguyên trong Private Subnet có thể kết nối Internet.

```
NATGatewayStack:
  Type: AWS::CloudFormation::Stack
  DependsOn: [PublicSubnet, InternetGatewayStack]
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/nat-gateway.yaml"
    Parameters:
      PublicSubnetID: !GetAtt PublicSubnet.Outputs.SubnetId
```

- Route Table: Định tuyến lưu lượng Internet.

```yaml
PrivateRouteTable:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/route-table-private.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      SubnetId: !GetAtt PrivateSubnet.Outputs.SubnetId
      NATGatewayId: !GetAtt NATGatewayStack.Outputs.NATGatewayID
      Name: private-rt

PublicRouteTable:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/route-table-public.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      SubnetId: !GetAtt PublicSubnet.Outputs.SubnetId
      InternetGatewayId: !GetAtt InternetGatewayStack.Outputs.InternetGatewayId
      Name: public-rt
```

- Security Group: kiểm soát lưu lượng vào/ra của các EC2 instances.

```yaml
SGPublicStack:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/security-group-public.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      AllowedSSHIp: !Ref AllowedSSHIp

SGPrivateStack:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/security-group-private.yaml"
    Parameters:
      VpcId: !GetAtt VPCStack.Outputs.VpcId
      PublicSecurityGroupId: !GetAtt SGPublicStack.Outputs.PublicSGId
```

- EC2 Instance: Nằm trong các Subnet tương ứng.

```yaml
PublicEC2Instance:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/ec2.yaml"
    Parameters:
      AMI: !Ref AMI
      SubnetId: !GetAtt PublicSubnet.Outputs.SubnetId
      InstanceType: !Ref InstanceType
      KeyName: !Ref KeyName
      SecurityGroupId: !GetAtt SGPublicStack.Outputs.PublicSGId
      AssociatePublicIp: true
      InstanceName: public-ec2

PrivateEC2Instance:
  Type: AWS::CloudFormation::Stack
  DependsOn: VPCStack
  Properties:
    TemplateURL: !Sub "https://${BucketName}.s3.amazonaws.com/CloudFormation/modules/ec2.yaml"
    Parameters:
      AMI: !Ref AMI
      SubnetId: !GetAtt PrivateSubnet.Outputs.SubnetId
      InstanceType: !Ref InstanceType
      KeyName: !Ref KeyName
      SecurityGroupId: !GetAtt SGPrivateStack.Outputs.PrivateSGId
      AssociatePublicIp: false
      InstanceName: private-ec2
```

3. Cấu hình tham số cho các dịch vụ

```
[
  {
    "ParameterKey": "KeyName",
    "ParameterValue": "lab1"
  },
  {
    "ParameterKey": "PublicSubnetCidr",
    "ParameterValue": "10.0.1.0/24"
  },
  {
    "ParameterKey": "PrivateSubnetCidr",
    "ParameterValue": "10.0.2.0/24"
  },
  {
    "ParameterKey": "VpcCidr",
    "ParameterValue": "10.0.0.0/16"
  },
  {
    "ParameterKey": "AvailabilityZone",
    "ParameterValue": "ap-southeast-1a"
  },
  {
    "ParameterKey": "AllowedSSHIp",
    "ParameterValue": "113.22.37.63/32"
  },
  {
    "ParameterKey": "InstanceType",
    "ParameterValue": "t3.micro"
  },
  {
    "ParameterKey": "AMI",
    "ParameterValue": "ami-05c261f9eb9de6a80"
  },
  {
    "ParameterKey": "BucketName",
    "ParameterValue": "cloudformation-bucket-2025"
  }
]
```

- Triển khai hạ tầng với CloudFormation

```
$ aws cloudformation create-stack \
  --stack-name lab1 \
  --template-url https://$BUCKET_NAME.s3.amazonaws.com/CloudFormation/main.yaml \
  --parameters file://CloudFormation/parameters.json \
  --capabilities CAPABILITY_IAM
{
    "StackId": "arn:aws:cloudformation:ap-southeast-1:715920390889:stack/lab1/f8fd5d70-2a5b-11f0-97a5-065463ab15c7"
}
```

III. Kết quả triển khai (xem chi tiết ở file README.md)

1. Terraform

```
=== Testing VPC and Networking ===

✅ VPC check (ID: vpc-069833fe845287414) successful
✅ Public Subnet check (ID: subnet-06e09c2dc1dba0b9c) successful
✅ Private Subnet check (ID: subnet-042c19c356b582726) successful
✅ Internet Gateway check (ID: igw-01f809493da26254f) successful
✅ NAT Gateway check (ID: nat-0821ab49627245ddc) successful

=== Testing Route Tables ===

✅ Public Route Table check (ID: rtb-082dc12f39d742ef9) successful
✅ Private Route Table check (ID: rtb-0ea9557c442ab9881) successful

=== Testing Security Groups ===

✅ Public Security Group check (ID: sg-0922de6585d01db9a) successful
✅ Private Security Group check (ID: sg-0edb72957451e9669) successful
✅ Default Security Group check (ID: sg-0e51d574646949141) successful

=== Testing EC2 Instances ===

✅ Public EC2 Instance check (ID: i-0d5f57b238563997b) successful
✅ Private EC2 Instance check (ID: i-0d62913442edab455) successful
```

- Kiểm tra Output:

Chạy lệnh:

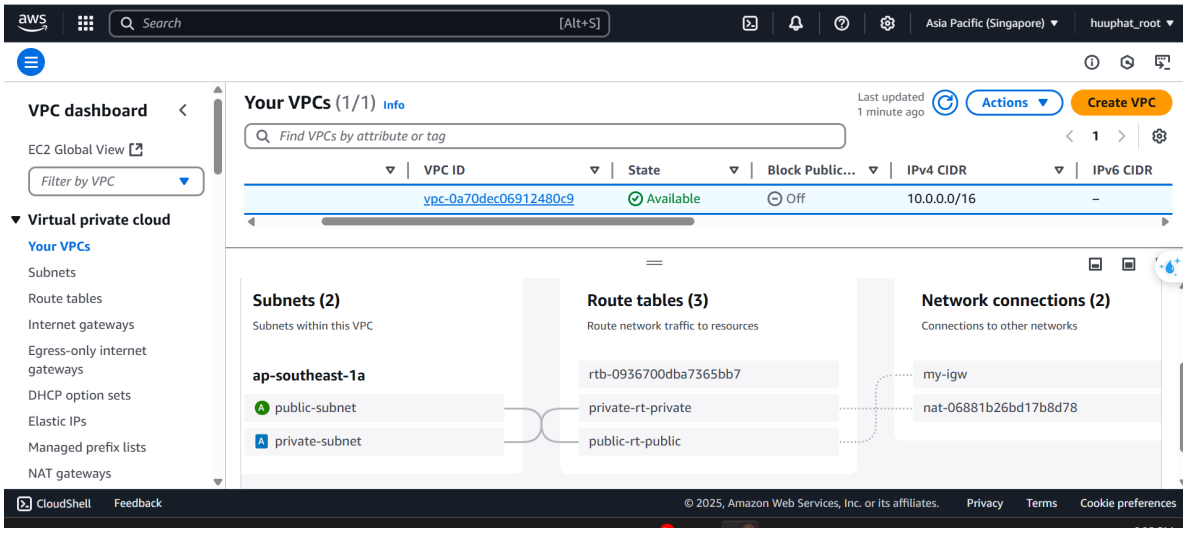
```
terraform init
terraform apply
```

Kết quả thu được:

```
Apply complete! Resources: 16 added, 0 changed, 0 destroyed.

Outputs:

nat_gateway_id = "nat-0ec24c61235e55f2f"
private_ec2_id = "i-0ade2445b426c13a2"
private_ec2_ip = "10.0.2.23"
public_ec2_id = "i-0cbecc4076392634c"
public_ec2_ip = "13.212.249.29"
vpc_id = "vpc-0729ad5dc45a39881"
```

Ở đây có thể return về id của các resources khác, nhưng quan trọng nhất là ip của puclic ec2 và private ec2 để thực hiện kiểm tra ssh.

- Vpc Source map:



2. CloudFormation

- Chạy file test_infra.sh với CloudFormation để kiểm tra từng dịch vụ đã được triển khai thành công:

```
=== Testing VPC and Networking ===

✅ VPC check (ID: vpc-069833fe845287414) successful
✅ Public Subnet check (ID: subnet-06e09c2dc1dba0b9c) successful
✅ Private Subnet check (ID: subnet-042c19c356b582726) successful
✅ Internet Gateway check (ID: igw-01f809493da26254f) successful
✅ NAT Gateway check (ID: nat-0821ab49627245ddc) successful

=== Testing Route Tables ===

✅ Public Route Table check (ID: rtb-082dc12f39d742ef9) successful
✅ Private Route Table check (ID: rtb-0ea9557c442ab9881) successful

=== Testing Security Groups ===

✅ Public Security Group check (ID: sg-0922de6585d01db9a) successful
✅ Private Security Group check (ID: sg-0edb72957451e9669) successful
✅ Default Security Group check (ID: sg-0e51d574646949141) successful

=== Testing EC2 Instances ===

✅ Public EC2 Instance check (ID: i-0d5f57b238563997b) successful
✅ Private EC2 Instance check (ID: i-0d62913442edab455) successful
```

- Output: Các output quan trọng như PublicEc2Ip và PrivateEc2Ip được trả về đầy đủ.

Outputs (9)

Search outputs

1

Key	Value	Description	Export name
InternetGatewayId	igw-052e4224a5a951cd8	Internet Gateway ID	-
NATGateWayId	nat-0b8caea1c817fed45	NAT Gateway ID	-
PrivateEc2Id	i-07915c7ca0bbc6274	-	-
PrivateEc2Ip	10.0.2.103	-	-
PrivateSubnetId	subnet-03936dea191ad1cc8	Private Subnet ID	-
PublicEc2Id	i-0a0e7bc57cf59f4e6	-	-
PublicEc2IP	3.1.213.89	-	-
PublicSubnetId	subnet-0e15ad42b8e614f74	Public Subnet ID	-
VpcId	vpc-06d5e203b2bd9d073	VPC ID	-

- CloudFormation Console:

Stack name	Status	Created time	Description
lab1-PrivateRouteTable-18XJ05WYML61M NESTED	CREATE_COMPLETE	2025-05-06 14:31:37 UTC+0700	Private Route Table Module
lab1-PrivateEC2Instance-QAVFAVD69OC5 NESTED	CREATE_COMPLETE	2025-05-06 14:29:33 UTC+0700	EC2 Instance
lab1-NATGatewayStack-EGU3MAF6IKH2 NESTED	CREATE_COMPLETE	2025-05-06 14:29:22 UTC+0700	NAT Gateway Module
lab1-PublicRouteTable-5RGWXFJJXV78 NESTED	CREATE_COMPLETE	2025-05-06 14:29:22 UTC+0700	Public Route Table Module
lab1-PublicEC2Instance-1CZA2WWPEGSR2 NESTED	CREATE_COMPLETE	2025-05-06 14:29:10 UTC+0700	EC2 Instance
lab1-SGPrivateStack-PQUG09JOYNWO NESTED	CREATE_COMPLETE	2025-05-06 14:29:10 UTC+0700	EC2 Instance
lab1-InternetGatewayStack-1HCI7WNY65NZA NESTED	CREATE_COMPLETE	2025-05-06 14:28:46 UTC+0700	Internet Gateway Module
lab1-PrivateSubnet-1WBB8SHLJ9HWB NESTED	CREATE_COMPLETE	2025-05-06 14:28:46 UTC+0700	Subnet Module
lab1-SGPublicStack-DTF9E2Y3RFHQ NESTED	CREATE_COMPLETE	2025-05-06 14:28:46 UTC+0700	EC2 Instance
lab1-PublicSubnet-1CVSVXJHU8FV3 NESTED	CREATE_COMPLETE	2025-05-06 14:28:46 UTC+0700	Subnet Module
lab1-VPCStack-15GD68JN0TK08 NESTED	CREATE_COMPLETE	2025-05-06 14:28:23 UTC+0700	VPC Module
lab1	CREATE_COMPLETE	2025-05-06 14:28:20 UTC+0700	Main stack for VPC Infrastructure

- Vcp Source map: Cho thấy các hạ tầng được liên kết với nhau một cách chính xác.

3. Kiểm tra kết nối vào các EC2 Instance

- SSH từ Local vào Public Ec2 instance:

```
(base) phatd@Phat-PC:~/devops/Lab1/Lab_1/Terraform$ ssh -o StrictHostKeyChecking=no -o ConnectTimeout=10 -i ./lab1.pem ec2-u
ser@54.169.57.60
Warning: Permanently added '54.169.57.60' (ED25519) to the list of known hosts.
       #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-10-0-1-19 ~]$
```

- SSH từ Public Ec2 instance vào Private Ec2 instance, kết quả thu được:


[ec2-user@ip-10-0-1-19 ~]$ ssh ec2-user@10.0.2.160
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-10-0-2-160 ~]$