58°

Lab - AWS re/Start

Automatización de implementaciones con AWS CloudFormation

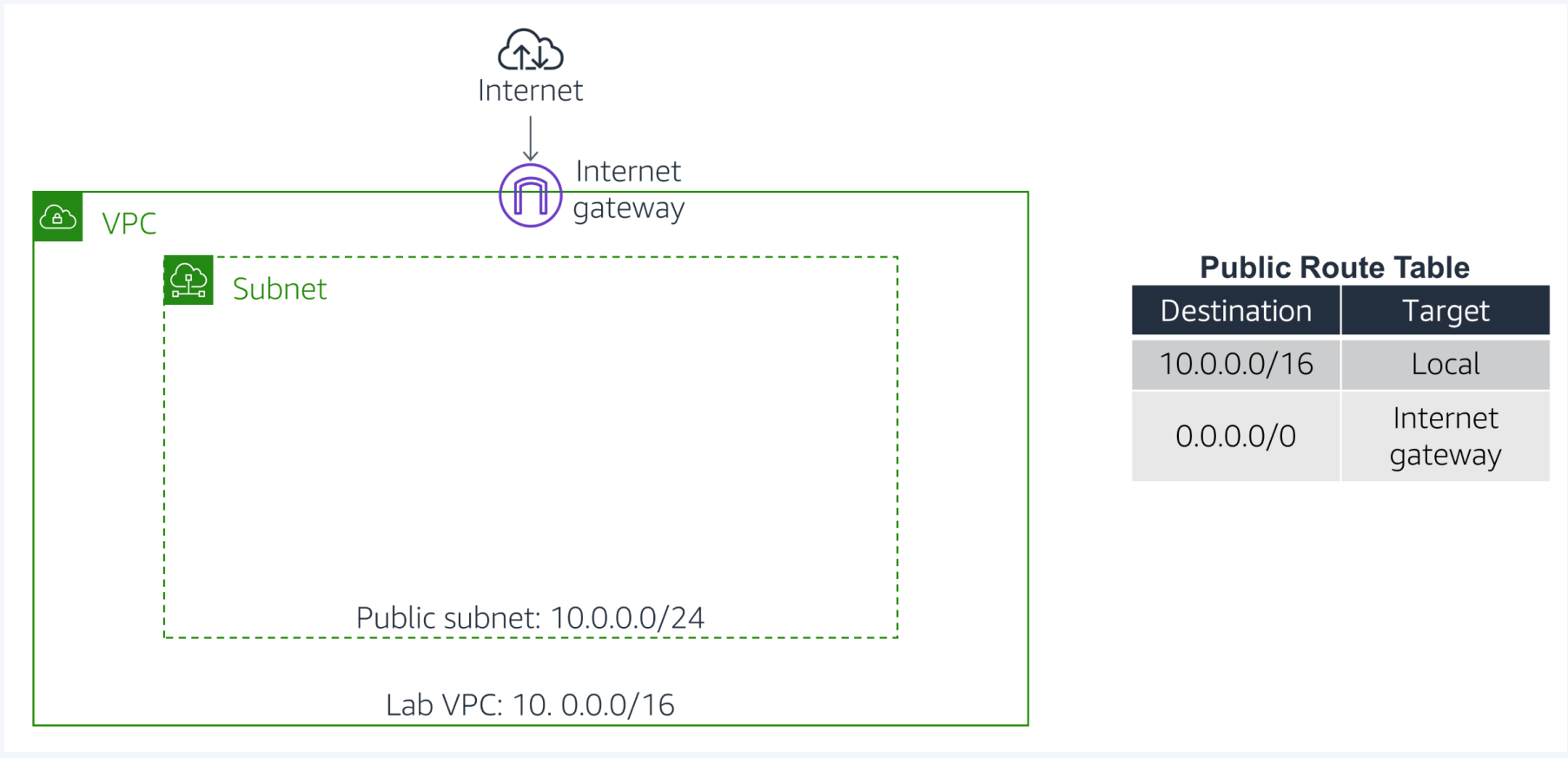
SWIPE

Tarea 01

Interactuando con AWS CloudWatch

Los objetivos son:

- Implementar una pila de AWS CloudFormation con una VPC y un grupo de seguridad definidos.
- Configurar una pila de AWS CloudFormation con recursos, como un bucket de S3 y EC2.
- Finalizar una pila de AWS CloudFormation y sus respectivos recursos.



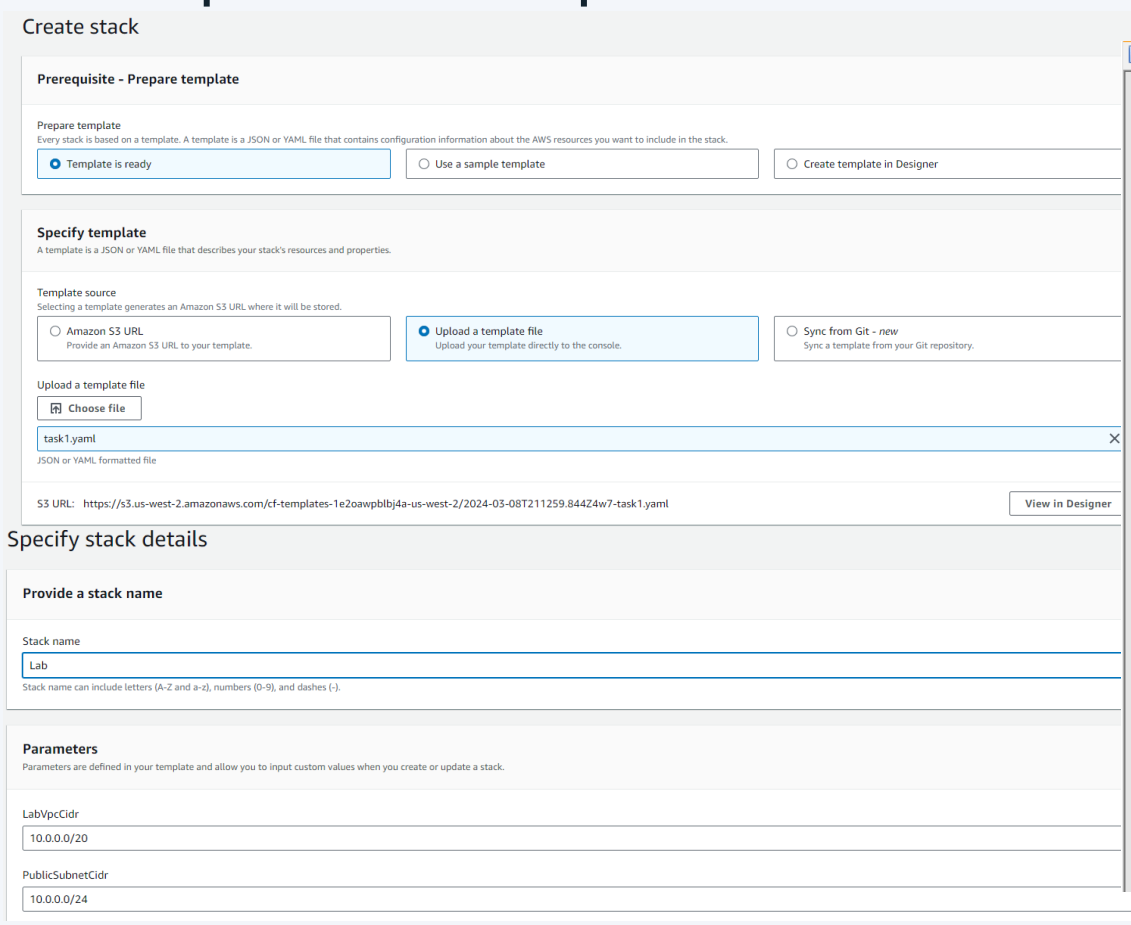
Public Route Table

Destination	Target
10.0.0.0/16	Local
0.0.0.0/0	Internet gateway

SWIPE

Tarea 01

Empezamos implementando una pila de CloudFormation

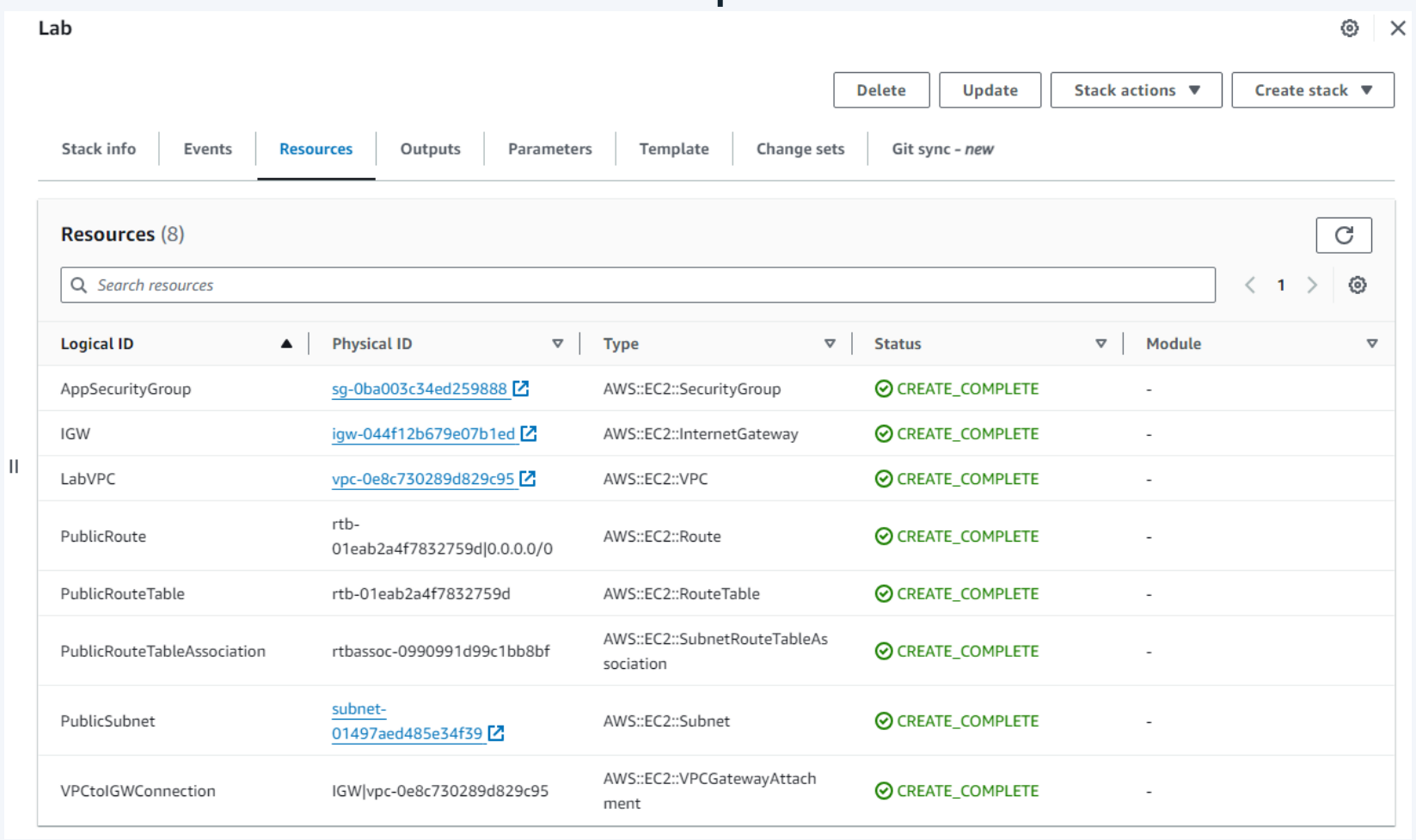


task1.yaml

```yaml
  LabVpcCidr:
    Type: String
    Default: 10.0.0.0/20

  PublicSubnetCidr:
    Type: String
    Default: 10.0.0.0/24


Resources:

############
# VPC with Internet Gateway
############

  LabVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: !Ref LabVpcCidr
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - Key: Name
          Value: Lab VPC

  IGW:
    Type: AWS::EC2::InternetGateway
    Properties:
      Tags:
        - Key: Name
          Value: Lab IGW

  VPCtoIGWConnection:
    Type: AWS::EC2::VPCGatewayAttachment
    DependsOn:
      - IGW
      - LabVPC
    Properties:
      InternetGatewayId: !Ref IGW
      VpcId: !Ref LabVPC
```

```yaml
############
# Public Route Table
############

  PublicRouteTable:
    Type: AWS::EC2::RouteTable
    DependsOn: LabVPC
    Properties:
      VpcId: !Ref LabVPC
      Tags:
        - Key: Name
          Value: Public Route Table

  PublicRoute:
    Type: AWS::EC2::Route
    DependsOn:
      - PublicRouteTable
      - IGW
    Properties:
      DestinationCidrBlock: 0.0.0.0/0
      GatewayId: !Ref IGW
      RouteTableId: !Ref PublicRouteTable

############
# Public Subnet
############

  PublicSubnet:
    Type: AWS::EC2::Subnet
    DependsOn: LabVPC
    Properties:
      VpcId: !Ref LabVPC
      MapPublicIpOnLaunch: true
      CidrBlock: !Ref PublicSubnetCidr
      AvailabilityZone: !Select
        - 0
        - !GetAZs
          Ref: AWS::Region
      Tags:
        - Key: Name
          Value: Public Subnet

  PublicRouteTableAssociation:
    Type: AWS::EC2::SubnetRouteTableAssociation
```

Esta ha sido la infraestructura que se levantó



Logical ID	Physical ID	Type	Status	Module
AppSecurityGroup	sg-0ba003c34ed259888	AWS::EC2::SecurityGroup	CREATE_COMPLETE	-
IGW	igw-044f12b679e07b1ed	AWS::EC2::InternetGateway	CREATE_COMPLETE	-
LabVPC	vpc-0e8c730289d829c95	AWS::EC2::VPC	CREATE_COMPLETE	-
PublicRoute	rtb-01eab2a4f7832759d|0.0.0.0/0	AWS::EC2::Route	CREATE_COMPLETE	-
PublicRouteTable	rtb-01eab2a4f7832759d	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PublicRouteTableAssociation	rtbassoc-0990991d99c1bb8bf	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-
PublicSubnet	subnet-01497aed485e34f39	AWS::EC2::Subnet	CREATE_COMPLETE	-
VPCtoIGWConnection	IGW|vpc-0e8c730289d829c95	AWS::EC2::VPCGatewayAttachment	CREATE_COMPLETE	-

SWIPE

Tarea 01

Ahora, procedemos a editar la plantilla con el fin de añadir un bucket dentro de la creación de infraestructura

Creating an Amazon S3 bucket with defaults

This example uses a AWS::S3::Bucket to create a bucket with default settings.

JSON

```
"myS3Bucket" : {
      "Type" : "AWS::S3::Bucket"
      }
```

YAML

```
MyS3Bucket:
    Type: AWS::S3::Bucket
```

Update stack

Prerequisite - Prepare template

Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

○ Use current template | ◉ Replace current template | ○ Edit template in designer

Specify template
A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

○ Amazon S3 URL | ◉ Upload a template file

Upload a template file
Choose file
task1.yaml

```yaml
Resources:
  MyS3Bucket:
    Type: AWS::S3::Bucket

    ###########
    # VPC with Internet Gateway
    ###########

  LabVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: !Ref LabVpcCidr
      EnableDnsSupport: true
      EnableDnsHostnames: true
      Tags:
        - Key: Name
          Value: Lab VPC
```

LabVPC	vpc-0e8c730289d829c95	AWS::EC2::VPC	CREATE_COMPLETE
MyS3Bucket	lab-mys3bucket-glf9ofz9ryvl	AWS::S3::Bucket	CREATE_IN_PROGRESS
PublicRoute	rtb-01eab2a4f7832759d|0.0.0.0/0	AWS::EC2::Route	CREATE_COMPLETE

También añadiremos una instancia de EC2

```yaml
Parameters:

  LabVpcCidr:
    Type: String
    Default: 10.0.0.0/20

  PublicSubnetCidr:
    Type: String
    Default: 10.0.0.0/24

  AmazonLinuxAMIID:
    Type: AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>
    Default: /aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2
```

```yaml
  MyEC2Instance:
    Type: AWS::EC2::Instance
    Properties:
      ImageID: !Ref AmazonLinuxAMIID
      InstanceType: t3.micro
      SecurityGroupIds:
       - !Ref AppSecurityGroup
      SubnetId: !Ref PublicSubnet
      Tags:
       - Key: Name
         Value: App Server
```

Y finalmente, eliminamos el stack (pila) de CloudFormation

Delete stack?

Delete stack **Lab** permanently? This action cannot be undone.

Deleting this stack will delete all stack resources. Resources will be deleted according to their DeletionPolicy. Learn more

Cancel | Delete

SWIPE