



Amazon Web Services
Build VPC With AWS CloudFormation

May 2020

Table of Contents

Resumen	3
Explorar la plantilla inicial de la VPC.....	3
Crear Stack	4
Laboratorio:	8
Referencias:	9
Actualiza tu Stack:.....	10
Appendice:.....	12

Resumen

Este laboratorio guiará al usuario a través del uso de AWS CloudFormation para crear una VPC con subredes públicas y privadas, describirá cada uno de los objetos creados por AWS CloudFormation y lanzará VPC con las subredes de VPC públicas y privadas, RouteTable, Elastic IP NAT Gateway, y un Bucket de S3.

El siguiente es un resumen del laboratorio:

- Explorar la plantilla inicial de CloudFormation
- Explorar los diferentes objetos de la VPC y lo que significan
- Lanzar un Stack de CloudFormation desde la Consola de AWS.
- Exportar el ID de la VPC y de los NAT Gateway junto con la URL del Bucket S3 en el "Output Tab"

El laboratorio proporcionará una plantilla inicial para que los usuarios exploren, después de crear el stack de la VPC a partir de una plantilla inicial, los usuarios deben completar el objetivo proporcionado para lograr la solución final.

Note: Se proporcionan capturas de pantalla para guiarlo a través de los pasos en el laboratorio. Los elementos que creará (por ejemplo, VPC, NAT Gateway, EIP) serán exclusivos de su cuenta, por lo que cosas como el ID de VPC que ve en la consola no necesariamente reflejarán lo que se ve en la captura de pantalla.

Explorar la plantilla inicial de la VPC

Explore el archivo de plantilla de AWS CloudFormation inicial. Puede usar cualquier editor de texto para explorar los diferentes elementos de VPC mencionados en la plantilla:

Lab_Initial_CloudFormation_Module_General_ImmersionDay.yml (El Código de la plantilla inicial se encuentra al final de la guía en el *Apendice*)

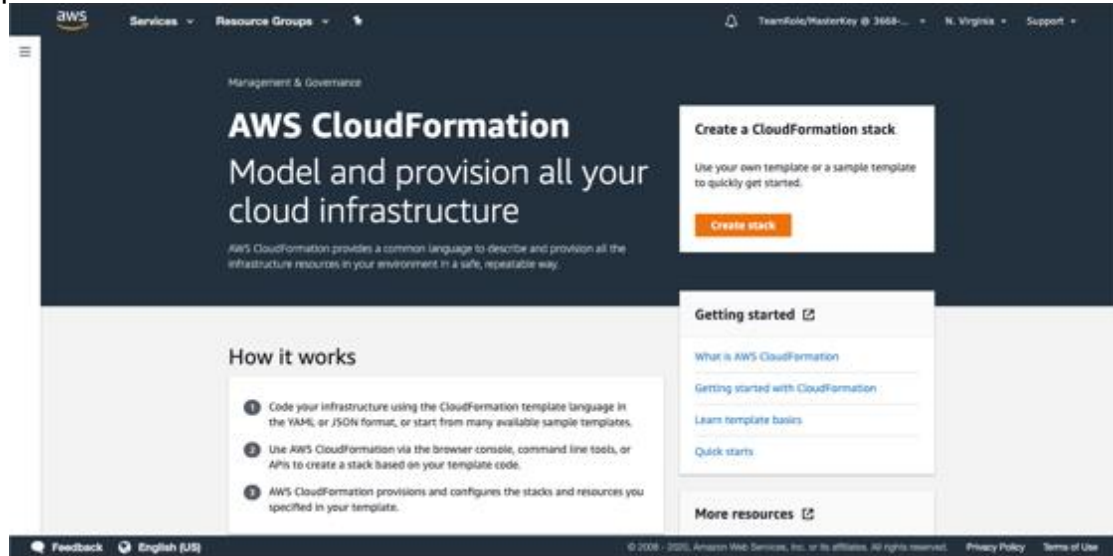
Notará los siguientes recursos en la plantilla inicial de AWS CloudFormation:

- VPC
- Internet Gateway
- S3 bucket
- Dos public subnets con sus tablas de ruteo
- Dos private subnets con sus tablas de ruteo
- Dos Elastic IP
- Dos NAT Gateway

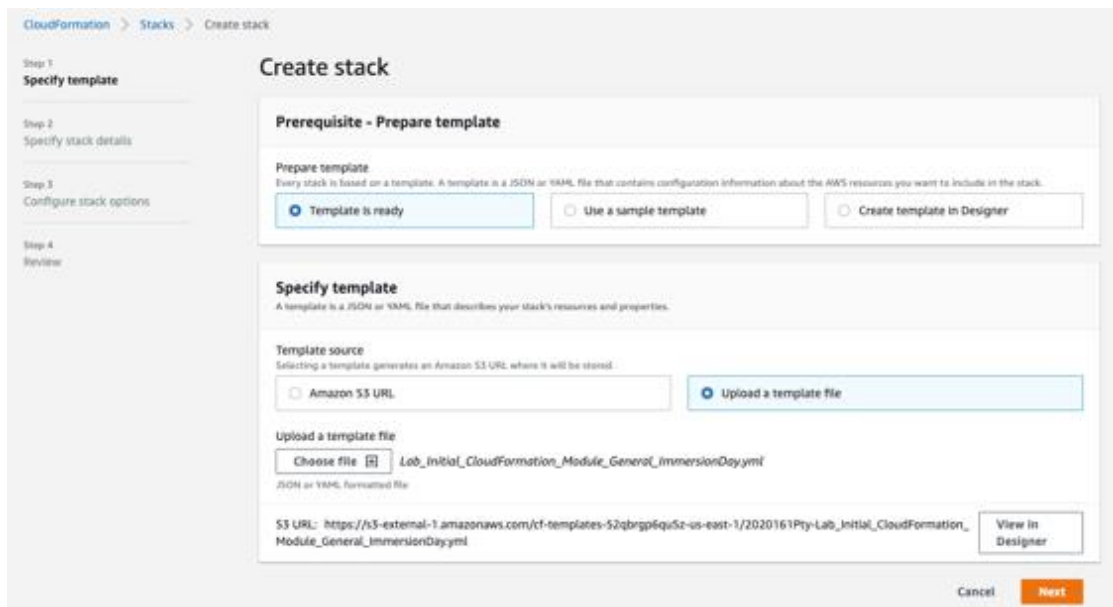
Build VPC with AWS CloudFormation Lab

Crear Stack

Ingresa a la **Consola de AWS**, y da click en **CloudFormation**, verás la siguiente pantalla:



Ahora da click en **Create stack** y en la parte inferior selecciona la opción **Upload a template file** y selecciona el template *Lab_Initial_CloudFormation_Module_General_ImmersionDay.yml*, finalmente da click en **Next**:



Build VPC with AWS CloudFormation Lab

En la siguiente pantalla da un nombre al Stack (por ejemplo *demo-vpc*). Recuerda que el nombre de tu stack debe ser único en la cuenta. Deja todas las otras opciones con los valores por defecto y da click en **Next**

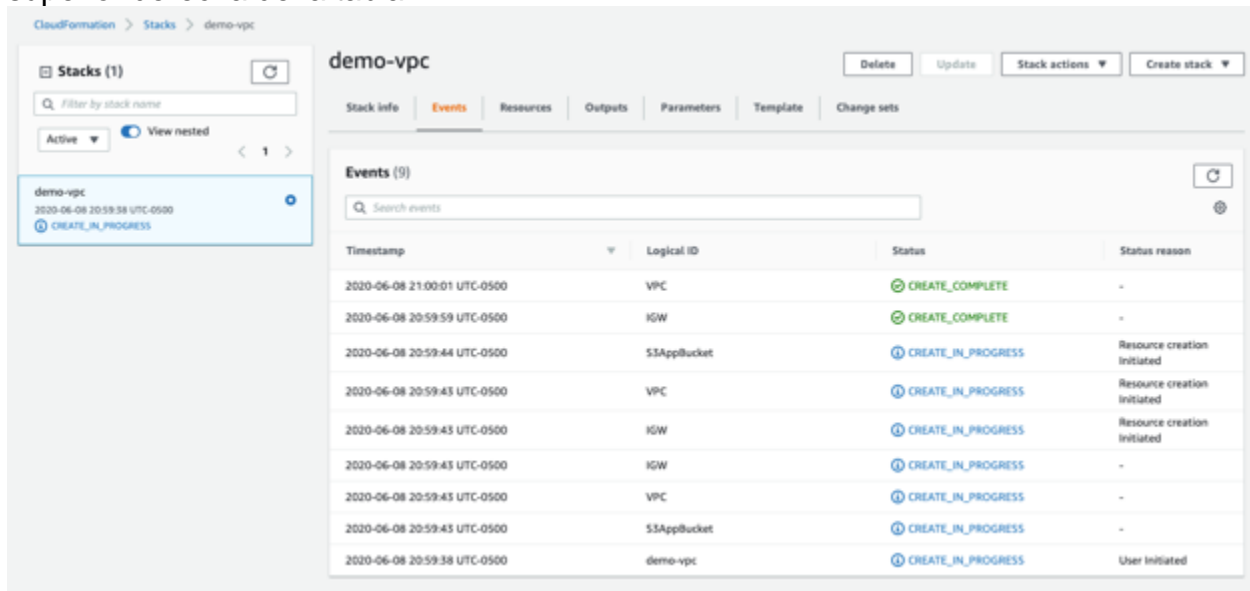
The screenshot shows the 'Specify stack details' step in the AWS CloudFormation console. On the left, a sidebar lists four steps: Step 1 (Specify template), Step 2 (Specify stack details), Step 3 (Configure stack options), and Step 4 (Review). The main area is titled 'Specify stack details'. It contains a 'Stack name' field with the value 'demo-vpc' and a note: 'Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-)'. Below this is a 'Parameters' section with three input fields: 'psharedacidr' (10.20.0.0/22), 'psharedbcidr' (10.20.4.0/22), and 'vpccidr' (10.20.0.0/16). At the bottom right are 'Cancel', 'Previous', and 'Next' buttons.

Acá puedes definir *tags* para tu Stack, un IAM Role y algunas otras opciones avanzadas como la protección de terminación de una instancia y/o el trigger del rollback. En este caso dejar las opciones como están y da click en **Next** de nuevo:

The screenshot shows the 'Configure stack options' step in the AWS CloudFormation console. The sidebar on the left highlights Step 3 (Configure stack options). The main area is titled 'Configure stack options'. It has three sections: 'Tags' (with a table for Key and Value, and an 'Add tag' button), 'Permissions' (with an 'IAM role - optional' dropdown set to 'Sample-role-name'), and 'Advanced options' (with expandable sections for 'Stack policy', 'Rollback configuration', 'Notification options', and 'Stack creation options'). At the bottom right are 'Cancel', 'Previous', and 'Next' buttons.

Ahora revisa la configuración de Stack y finalmente da click en **Create Stack** al final de la pantalla.

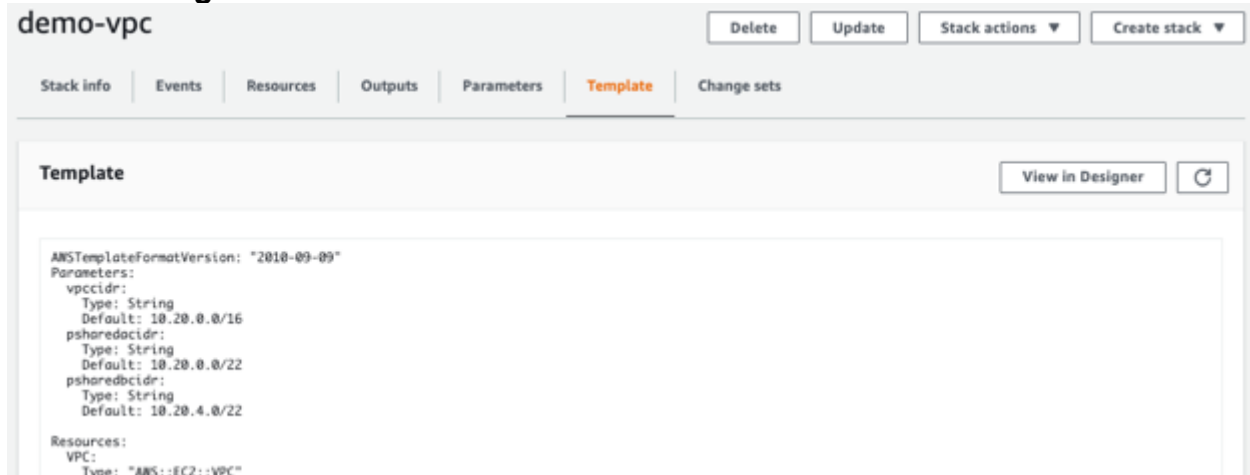
Una vez inicia la creación del Stack, eres enviado a la sección de eventos del Stack, puedes refrescarlos dando click sobre el botón de refrescar (🔄) en la esquina superior derecha de la tabla.



The screenshot shows the AWS CloudFormation console for a stack named 'demo-vpc'. The 'Events' tab is selected, displaying a list of events. The stack is in the 'CREATE_IN_PROGRESS' state. The events table shows the following details:

Timestamp	Logical ID	Status	Status reason
2020-06-08 21:00:01 UTC-0500	VPC	CREATE_COMPLETE	-
2020-06-08 20:59:59 UTC-0500	IGW	CREATE_COMPLETE	-
2020-06-08 20:59:44 UTC-0500	S3AppBucket	CREATE_IN_PROGRESS	Resource creation initiated
2020-06-08 20:59:43 UTC-0500	VPC	CREATE_IN_PROGRESS	Resource creation initiated
2020-06-08 20:59:43 UTC-0500	IGW	CREATE_IN_PROGRESS	Resource creation initiated
2020-06-08 20:59:43 UTC-0500	IGW	CREATE_IN_PROGRESS	-
2020-06-08 20:59:43 UTC-0500	VPC	CREATE_IN_PROGRESS	-
2020-06-08 20:59:43 UTC-0500	S3AppBucket	CREATE_IN_PROGRESS	-
2020-06-08 20:59:38 UTC-0500	demo-vpc	CREATE_IN_PROGRESS	User initiated

Mientras esperas puedes explorar las otras pestañas del Stack, como por ejemplo la pestaña de “Parameters” donde encontrarás los parámetros que se ingresaron en la creación del Stack o la pestaña de “Template” donde encontrarás el código de la plantilla y donde podrás ingresar al Designer de Cloudformation dando click en el botón **View in Designer**



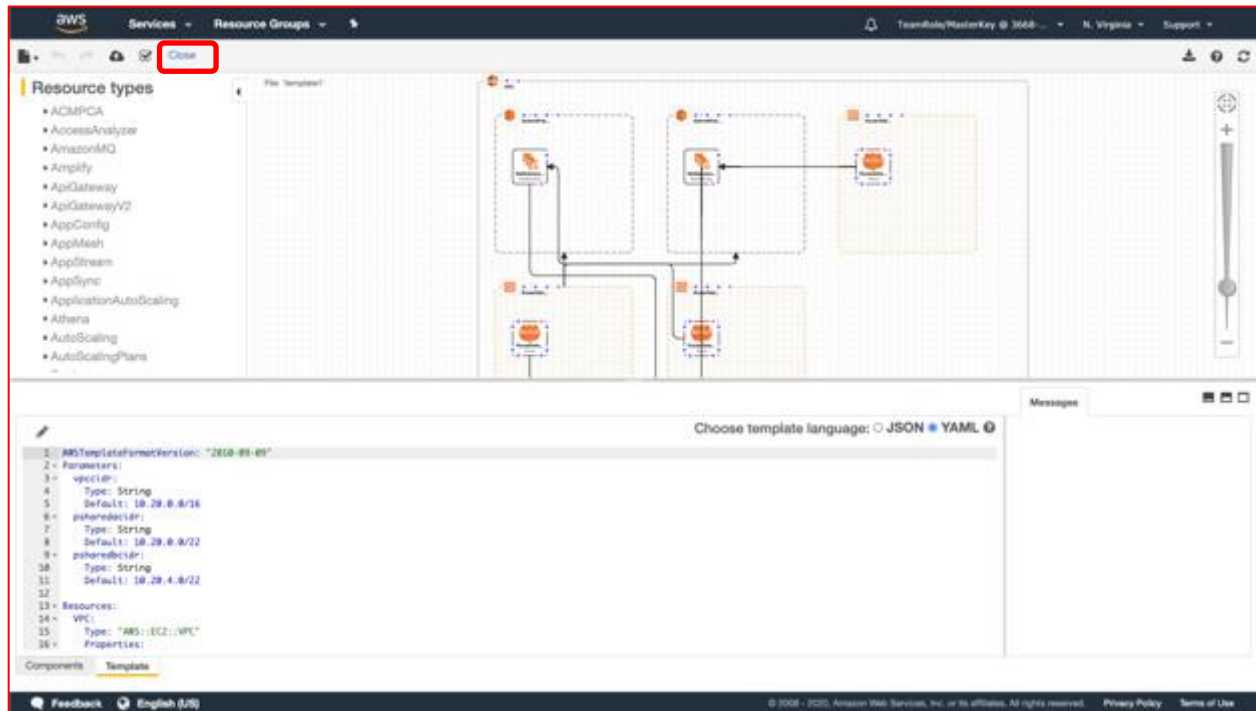
The screenshot shows the AWS CloudFormation console for a stack named 'demo-vpc'. The 'Template' tab is selected, displaying the template code. The code defines parameters for VPC, IGW, and S3 buckets, and resources for VPC, IGW, and S3 buckets. The 'View in Designer' button is visible in the top right corner.

```
AWSTemplateFormatVersion: "2010-09-09"
Parameters:
  vpcid:
    Type: String
    Default: 10.20.0.0/16
  psharedid:
    Type: String
    Default: 10.20.0.0/22
  psharedbid:
    Type: String
    Default: 10.20.4.0/22
Resources:
  VPC:
    Type: "AWS::EC2::VPC"
```

AWS CloudFormation Designer (Designer) es una herramienta gráfica para crear, consultar y modificar las plantillas de AWS CloudFormation. Con Designer, puedes hacer un diagrama de los recursos de la plantilla con una interfaz de arrastrar y soltar y, a continuación, editar los detalles mediante editor de JSON y YAML integrado.

Build VPC with AWS CloudFormation Lab

Para salir del diseñador da click en el botón de **Close** en la parte superior.



Una vez el Status del Stack cambie a **CREATE_COMPLETE**, por favor ingresa a la pestaña de **Resources** para ver todos los recursos creados por esta plantilla.

demo-vpc

Delete Update Stack actions Create stack

Stack info Events **Resources** Outputs Parameters Template Change sets

Resources (19)

Search resources

Logical ID	Physical ID	Type	Status	Status reason
BucketPolicyApp	demo-vpc-BucketPolicyApp-NFYB87HQ7KNG	AWS::S3::BucketPolicy	CREATE_COMPLETE	-
EIPNatGWA	54.237.5.3	AWS::EC2::EIP	CREATE_COMPLETE	-
EIPNatGWB	54.144.124.103	AWS::EC2::EIP	CREATE_COMPLETE	-
GatewayAttach	demo-Gatew-1RHY12IS9YX0H	AWS::EC2::VPCGatewayAttachment	CREATE_COMPLETE	-
IGW	igw-0f46443f0a653bb87	AWS::EC2::InternetGateway	CREATE_COMPLETE	-
NatGatewayA	nat-0d1eaab83f3c71584	AWS::EC2::NatGateway	CREATE_COMPLETE	-
NatGatewayB	nat-061210942131d3014	AWS::EC2::NatGateway	CREATE_COMPLETE	-
RouteDefaultPrivateA	demo-Route-RTQDYHBS6EU	AWS::EC2::Route	CREATE_COMPLETE	-
RouteDefaultPrivateB	demo-Route-G2HRV9PZ8C4U	AWS::EC2::Route	CREATE_COMPLETE	-
RouteDefaultPublic	demo-Route-1VZPFA3FFDYX	AWS::EC2::Route	CREATE_COMPLETE	-
RouteTablePrivateA	rtb-06087160c139b2bb1	AWS::EC2::RouteTable	CREATE_COMPLETE	-

Laboratorio:

Ahora, por favor modifica el template con los siguientes objetivos:

Agrega restricción en los parámetros:

- Vpccidr
 - El texto mínimo debe ser de 9 caracteres
 - El máximo debe ser de 18
 - El patrón permitido debe ser:
"(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})/(\d{1,2})"
 - Pon una descripción en la restricción
- Psharedacidr
 - El texto mínimo debe ser de 9 caracteres
 - El máximo debe ser de 18
 - El patrón permitido debe ser:
"(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})/(\d{1,2})"
 - Pon una descripción en la restricción
- Psharedbcidr
 - El texto mínimo debe ser de 9 caracteres
 - El máximo debe ser de 18
 - El patrón permitido debe ser:
"(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})\.\.(\d{1,3})/(\d{1,2})"
 - Pon una descripción en la restricción

Agrega una restricción de borrado:

- Cree una política de eliminación para que su Bucket de S3 se retenga (retain) en la eliminación

Agrega la sección de Outputs para mostrar valores en la pestaña de Output:

- Vpc id
 - Crear una descripción de tu output
 - Referencia tu VPC como el valor usando ¡Ref
- NATGWA
 - Crear una descripción de tu output
 - Referencia tu NAT gateway A como el valor usando ¡Ref
- NATGWB
 - Crear una descripción de tu output
 - Referencia tu NAT gateway B como el valor usando ¡Ref
- App bucket URL
 - Crear una descripción de tu output
 - Referencia la URL de tu Bucket S3 como el valor usando ¡Ref

Agrega valores de “Export” en la sección de Outputs para referencias Cross-Stack:

- Vpc id
 - Exporta tu VPC con el nombre ‘sharedinf-vpc’
- App bucket URL
 - Exporta la url de tu bucket S3 con el nombre ‘sharedinf-appbucketurl’

Referencias:

Parámetros:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/parameters-section-structure.html>

Intrinsic functions:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-function-reference.html>

Outputs y Export:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/outputs-section-structure.html>

Mappings:

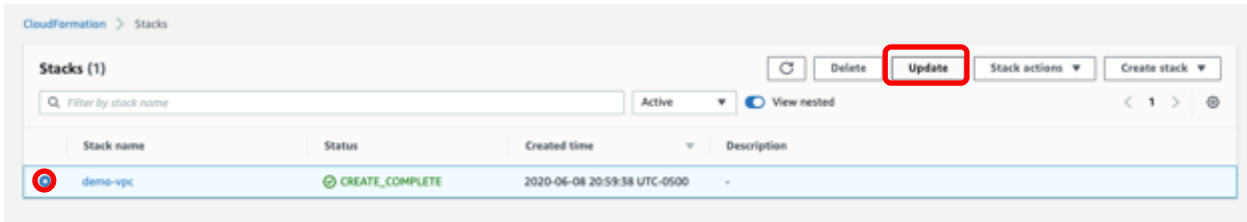
<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/mappings-section-structure.html>

Deletion policy:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-attribute-deletionpolicy.html>

Actualiza tu Stack:

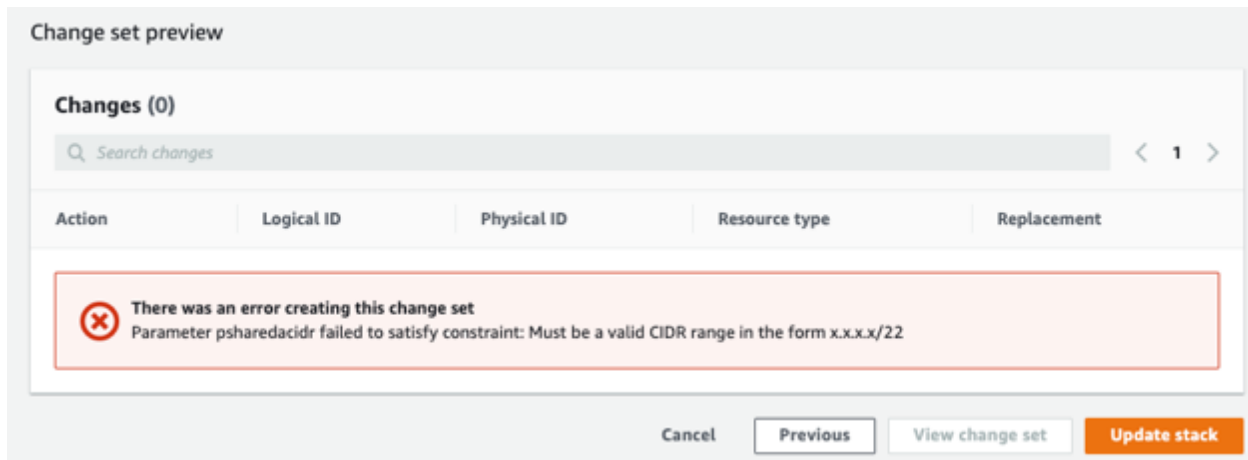
Una vez modificada tu plantilla existente, puedes usar la opción de **Update Stack** para actualizar tu Stack. Para esto, selecciona tu Stack y da click en el botón de **Update Stack** en la parte superior.



En la Ventana de **Update Stack**, selecciona la opción de **Replace current template** y después selecciona la opción **Upload a template file**, busca tu plantilla actualizada, súbela y da click en **Next**

The screenshot shows the 'Update stack' wizard. The first section is 'Prerequisite - Prepare template', which has three radio buttons: 'Use current template', 'Replace current template' (which is selected), and 'Edit template in designer'. The second section is 'Specify template', which has a 'Template source' section with two radio buttons: 'Amazon S3 URL' and 'Upload a template file' (which is selected). Below this is a file upload section with a 'Choose file' button and a text input field containing the filename 'Lab_Initial_CloudFormation_Module_General_ImmersionDay_2.yml'. Below the filename is the text 'JSON or YAML formatted file'. At the bottom of the wizard, there is an 'S3 URL' field and a 'View in Designer' button. The bottom right of the wizard has 'Cancel' and 'Next' buttons.

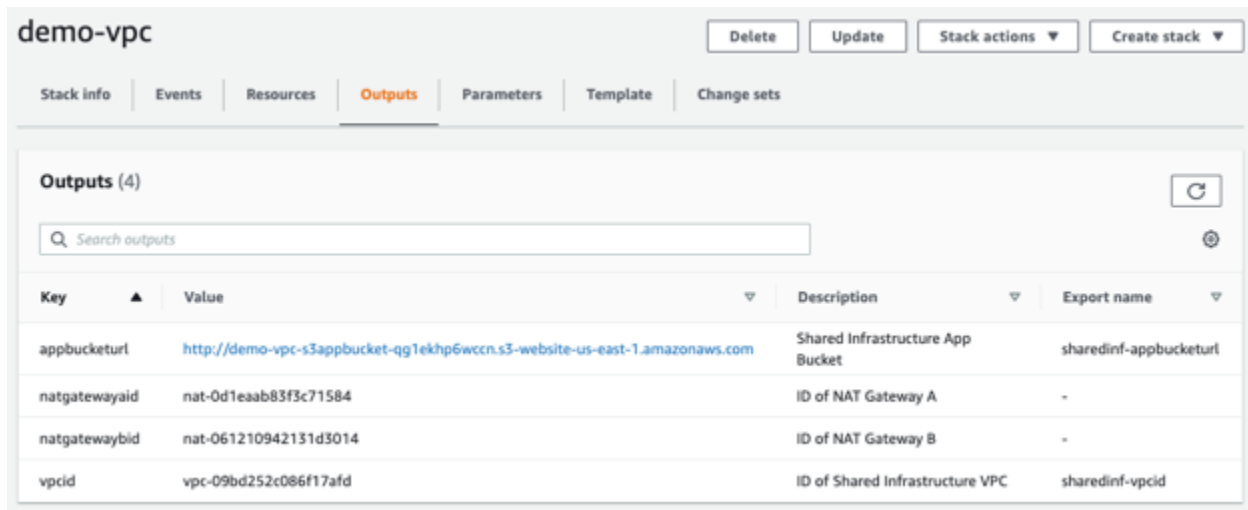
Ahora los pasos siguientes son los que seguiste al comienzo de la guía para crear el Stack. Puedes probar tus restricciones colocando un valor indebido en los parámetros y dando Click en **Next** hasta llegar al resumen del Stack:



Puedes dar click en **Previous** para corregir el valor.

Ahora verás que el Status de tu Stack es **UPDATE_IN_PROGRESS** y en la pestaña de **Events** verás la actividad que se está ejecutando.

Una vez el estado de tu Stack cambie a **UPDATE_COMPLETE**, puedes seleccionar la sección de **Outputs** y verás los valores del Output:



Also, click on **CloudFormation** icon on the right top corner of the screen and select **Exports** option, you will find two exported value shown in here which can be utilized for cross-stack reference.

Build VPC with AWS CloudFormation Lab



The screenshot shows the AWS CloudFormation console with the 'Exports' tab selected. A table lists two exports from the 'demo-vpc' stack:

Export Value	Stack Name	Stack ID
sharedinf-appbucketurl	demo-vpc	arn:aws:cloudformation:us-east-1:789211807855:stack/demo-vpc/58064a90-...
sharedinf-vpcid	demo-vpc	arn:aws:cloudformation:us-east-1:789211807855:stack/demo-vpc/58064a90-...

To create a cross-stack reference, use the **Export** output field to flag the value of a resource-output for export. Then, use the **Fn:: ImportValue** intrinsic function to import the value.

Appendix:

Initial AWS CloudFormation Template for lab exercise:

Create a file `Lab_Initial_CloudFormation_Module_General_ImmersionDay.yaml` and copy paste following code:

```
AWSTemplateFormatVersion: '2010-09-09'
Parameters:
  vpccidr:
    Type: String
    Default: 10.20.0.0/16
  psharedacidr:
    Type: String
    Default: 10.20.0.0/22
  psharedbcidr:
    Type: String
    Default: 10.20.4.0/22

Resources:
  VPC:
    Type: "AWS::EC2::VPC"
    Properties:
      CidrBlock: !Ref vpccidr
  IGW:
    Type: "AWS::EC2::InternetGateway"
  S3AppBucket:
    Type: "AWS::S3::Bucket"
    Properties:
      AccessControl: PublicRead
      WebsiteConfiguration:
        ErrorDocument: index.html
        IndexDocument: index.html
```

```

BucketPolicyApp:
  Type: "AWS::S3::BucketPolicy"
  Properties:
    Bucket: !Ref S3AppBucket
    PolicyDocument:
      Statement:
        -
          Sid: "ABC123"
          Action:
            - "s3:GetObject"
          Effect: Allow
          Resource: !Join ["", ["arn:aws:s3:::", !Ref S3AppBucket, "/*"]]
          Principal:
            AWS:
              - "*"

GatewayAttach:
  Type: "AWS::EC2::VPCGatewayAttachment"
  Properties:
    InternetGatewayId: !Ref IGW
    VpcId: !Ref VPC

SubnetPublicSharedA:
  Type: "AWS::EC2::Subnet"
  Properties:
    AvailabilityZone: !Select [0, !GetAZs ]
    CidrBlock: !Ref psharedacidr
    MapPublicIpOnLaunch: true
    VpcId: !Ref VPC

SubnetPublicSharedB:
  Type: "AWS::EC2::Subnet"
  Properties:
    AvailabilityZone: !Select [1, !GetAZs ]
    CidrBlock: !Ref psharedbcidr
    MapPublicIpOnLaunch: true
    VpcId: !Ref VPC

SubnetRouteTableAssociatePublicA:
  Type: "AWS::EC2::SubnetRouteTableAssociation"
  Properties:
    RouteTableId: !Ref RouteTablePublic
    SubnetId: !Ref SubnetPublicSharedA

SubnetRouteTableAssociatePublicB:
  Type: "AWS::EC2::SubnetRouteTableAssociation"
  Properties:
    RouteTableId: !Ref RouteTablePublic
    SubnetId: !Ref SubnetPublicSharedB

RouteDefaultPublic:
  Type: "AWS::EC2::Route"

```

```

DependsOn: GatewayAttach
Properties:
  DestinationCidrBlock: 0.0.0.0/0
  GatewayId: !Ref IGW
  RouteTableId: !Ref RouteTablePublic
RouteDefaultPrivateA:
  Type: "AWS::EC2::Route"
  Properties:
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NatGatewayA
    RouteTableId: !Ref RouteTablePrivateA
RouteDefaultPrivateB:
  Type: "AWS::EC2::Route"
  Properties:
    DestinationCidrBlock: 0.0.0.0/0
    NatGatewayId: !Ref NatGatewayB
    RouteTableId: !Ref RouteTablePrivateB
RouteTablePublic:
  Type: "AWS::EC2::RouteTable"
  Properties:
    VpcId: !Ref VPC
RouteTablePrivateA:
  Type: "AWS::EC2::RouteTable"
  Properties:
    VpcId: !Ref VPC
RouteTablePrivateB:
  Type: "AWS::EC2::RouteTable"
  Properties:
    VpcId: !Ref VPC
EIPNatGWA:
  DependsOn: GatewayAttach
  Type: "AWS::EC2::EIP"
  Properties:
    Domain: vpc
EIPNatGWB:
  DependsOn: GatewayAttach
  Type: "AWS::EC2::EIP"
  Properties:
    Domain: vpc
NatGatewayA:
  Type: "AWS::EC2::NatGateway"
  Properties:
    AllocationId: !GetAtt EIPNatGWA.AllocationId
    SubnetId: !Ref SubnetPublicSharedA
NatGatewayB:
  Type: "AWS::EC2::NatGateway"

```

Properties:

AllocationId: !GetAtt EIPNatGWB.AllocationId

SubnetId: !Ref SubnetPublicSharedB