

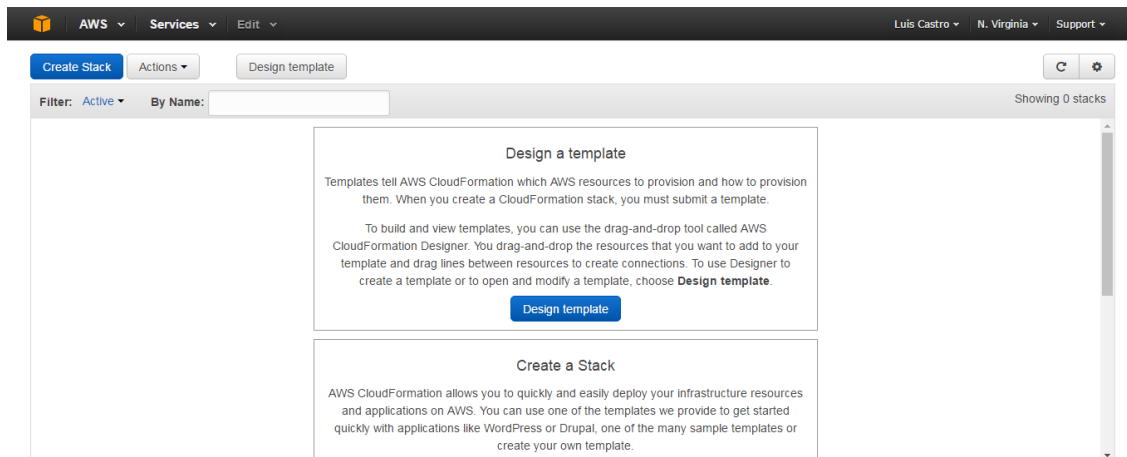
Step 1

Access the AWS console through the following link:

<https://450006219561.signin.aws.amazon.com/console>

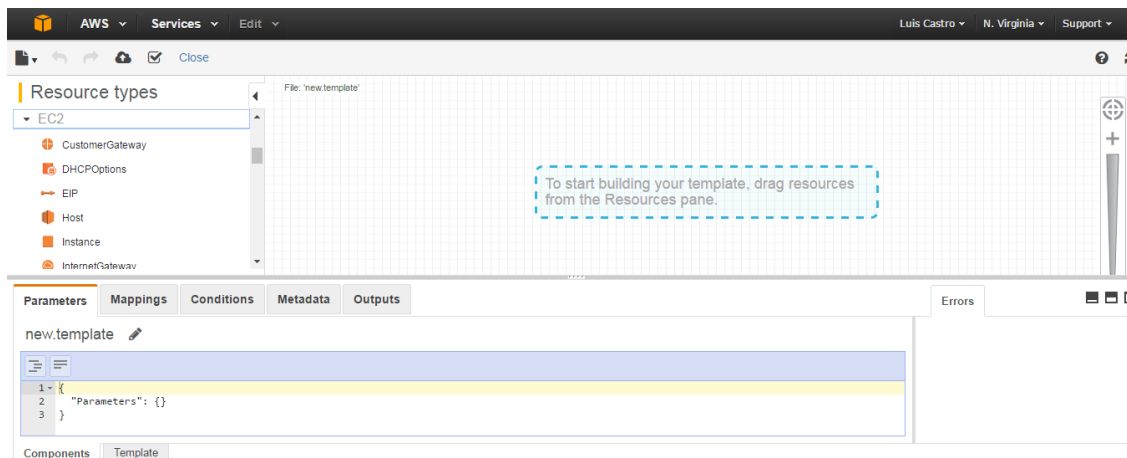
Step 2

Enter the CloudFormation service and choose Design Template

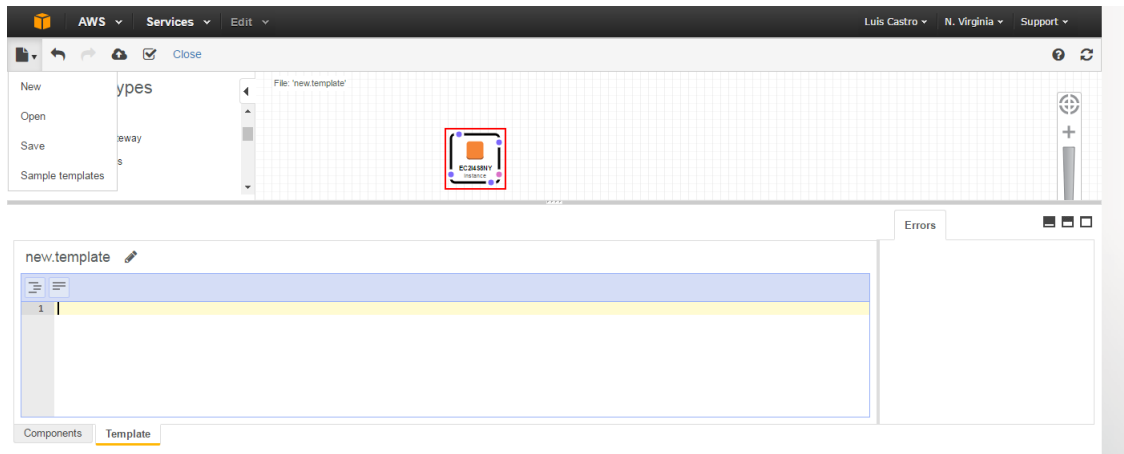


Step 3

Within Resource Type choose EC2> Instance and drag the icon to the panel



Mark the instance and in the menu below in Properties select Template and delete the existing template

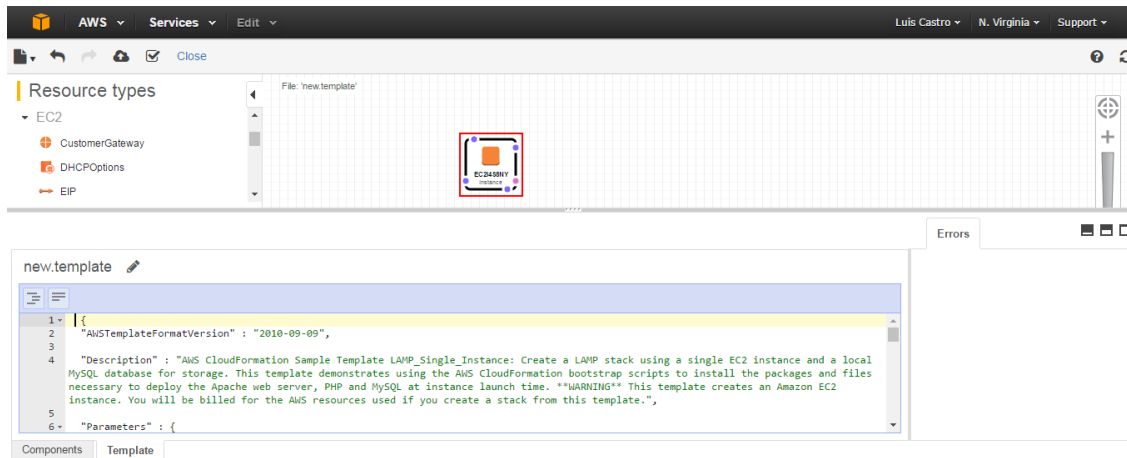


Step 5

From the file sent by mail use the call:

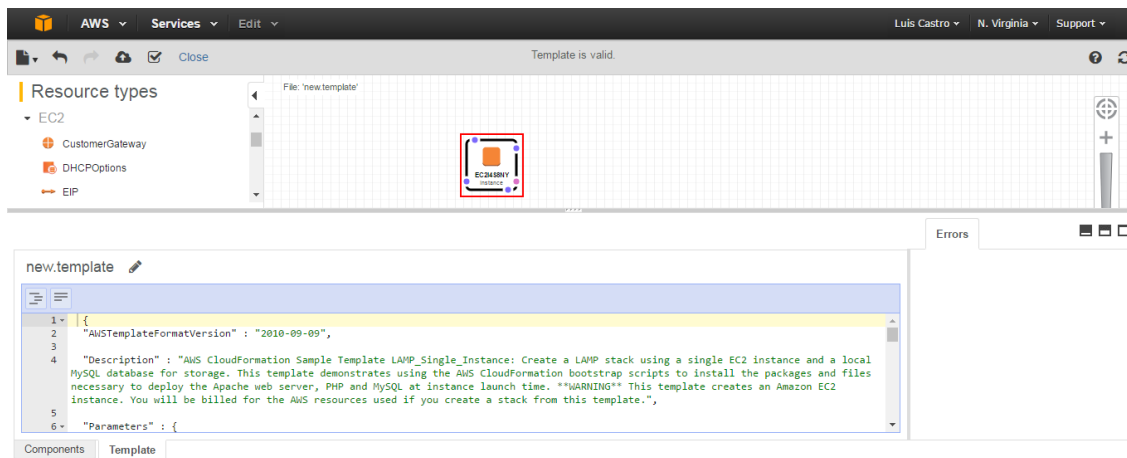
- EC2_SG.json

Copy the code and paste it back into the Template field



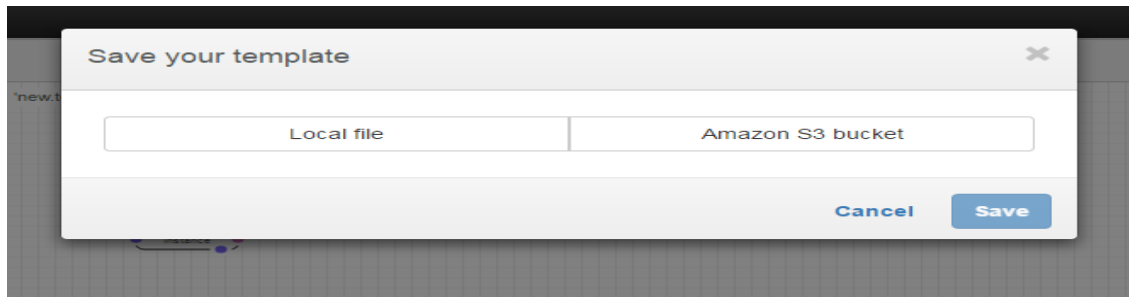
Step 6

Click on the Check found in the main bar "Validate Template" and wait for the Template validation to run until "Template is Valid" appears



Step 7

In the main menu click on the icon in the form of a leaf and enter save to save the template with the name EC2_SG_2.json



Save your template

Local file

Amazon S3 bucket

File name:

EC2_SG_2|template

Cancel

Save

Step 8

Click on the cloud icon in the main bar called Create Stack

Find the previously created file EC2_SG_2.template and click Next.

CloudFormation > Stacks > Create stack

Step 1

Specify template

Step 2

Specify stack details

Step 3

Configure stack options

Step 4

Review

Create 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.

☒ Template is ready

☐ Use a sample template

☐ Create 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

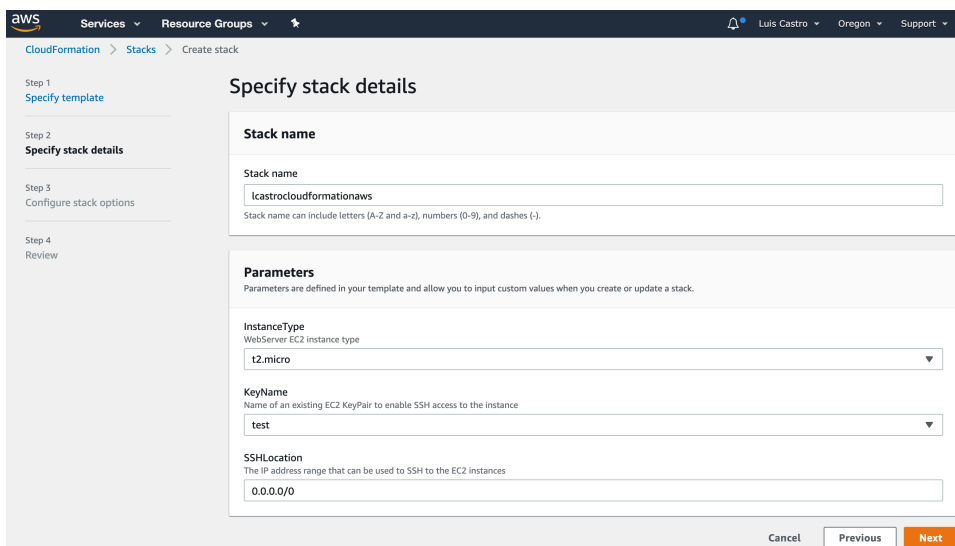
Choose file

EC2_SG_2.template

JSON or YAML formatted file

Step 9

- **Stack Name**
 - Username + cloudformationaws
 - **Ex: lcastrocloudformationaws**
- **Instance Type**
 - T2.small (Verify that by default the selected instance is T2.small and change it to t2.micro)
- **KeyName**
 - The SSH key used in the previous lab



aws Services Resource Groups Luis Castro Oregon Support

CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Specify stack details

Stack name

Stack name

lcastrocloudformationaws

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

InstanceType

WebServer EC2 instance type

t2.micro

KeyName

Name of an existing EC2 KeyPair to enable SSH access to the instance

test

SSHLocation


The IP address range that can be used to SSH to the EC2 instances

0.0.0.0/0

Cancel Previous Next

Step 10

- **Key**
 - o Name
- **Value**
 - o username + cloudformationsmall
 - Ex: lcastrocloudformationsmall
- **Create**



AWS

Services

Edit

Luis Castro

N. Virginia

Support

Create stack

Select Template

Specify Details

Options

Review

Options

Tags

You can specify tags (key-value pairs) for resources in your stack. You can add up to 10 unique key-value pairs for each stack. [Learn more.](#)

	Key (127 characters maximum)	Value (255 characters maximum)	
1	name	lcastrocloudformationsmall	+

Advanced

You can set additional options for your stack, like notification options and a stack policy. [Learn more.](#)

Cancel

Previous

Next

Options

Tags

name

lcastrocloudformationsmall

Advanced

Notification

Timeout

none

Rollback on failure

Yes

Cancel

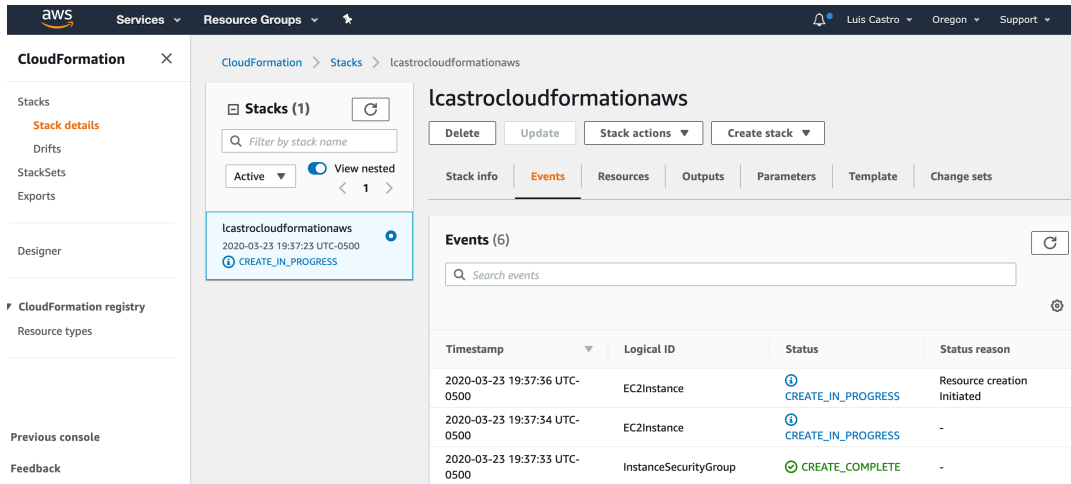
Previous

Create

400.753.4000 | 1000 E. Bay Street, Suite 200 | Palo Alto, CA 94301

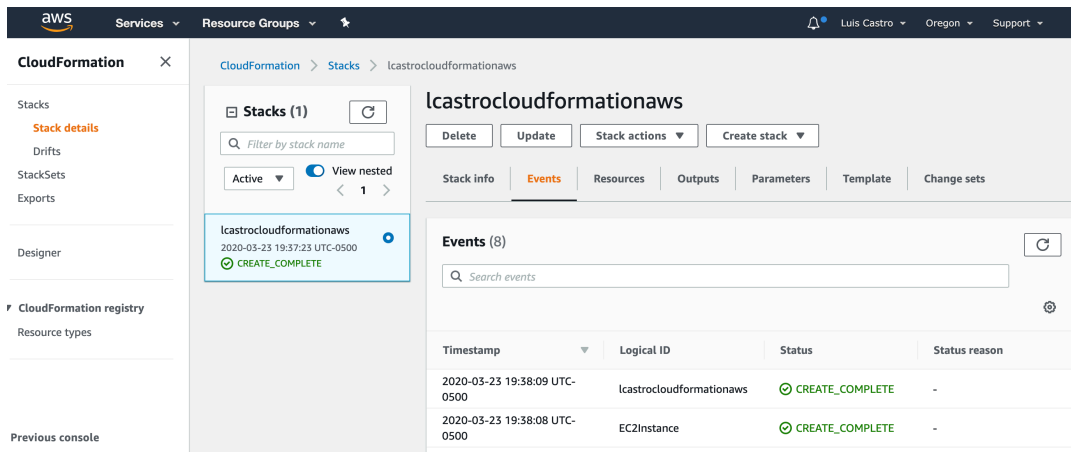
Step 11

- Validate that the stack is being created and that an EC2 machine is being created



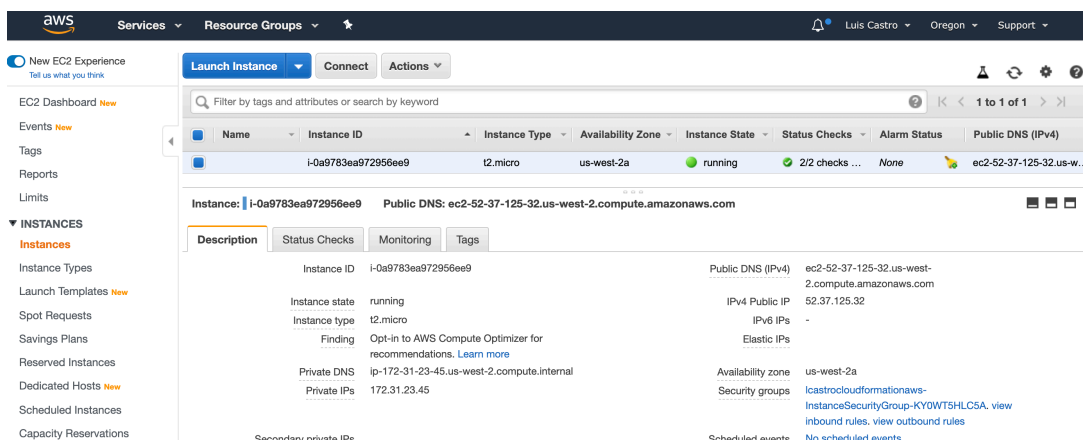
The screenshot shows the AWS CloudFormation console for the stack 'lcastrocloudformationaws'. The stack is in the 'CREATE_IN_PROGRESS' state. The 'Events' tab is selected, showing a list of events:

Timestamp	Logical ID	Status	Status reason
2020-03-23 19:37:36 UTC-0500	EC2Instance	CREATE_IN_PROGRESS	Resource creation Initiated
2020-03-23 19:37:34 UTC-0500	EC2Instance	CREATE_IN_PROGRESS	-
2020-03-23 19:37:33 UTC-0500	InstanceSecurityGroup	CREATE_COMPLETE	-



The screenshot shows the AWS CloudFormation console for the stack 'lcastrocloudformationaws'. The stack is now in the 'CREATE_COMPLETE' state. The 'Events' tab is selected, showing a list of events:

Timestamp	Logical ID	Status	Status reason
2020-03-23 19:38:09 UTC-0500	lcastrocloudformationaws	CREATE_COMPLETE	-
2020-03-23 19:38:08 UTC-0500	EC2Instance	CREATE_COMPLETE	-



The screenshot shows the AWS EC2 console for the instance 'i-0a9783ea972956ee9'. The instance is in the 'running' state. The 'Description' tab is selected, showing the following details:

Field	Value
Instance ID	i-0a9783ea972956ee9
Instance state	running
Instance type	t2.micro
Finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Private DNS	ip-172-31-23-45.us-west-2.compute.internal
Private IPs	172.31.23.45
Secondary private IPs	-
Public DNS (IPv4)	ec2-52-37-125-32.us-west-2.compute.amazonaws.com
IPv4 Public IP	52.37.125.32
IPv6 IPs	-
Elastic IPs	-
Availability zone	us-west-2a
Security groups	lcastrocloudformationaws-InstanceSecurityGroup-KYOWTSHLC5A. view inbound rules , view outbound rules
Scheduled events	No scheduled events

Step 12

- Find the previously created file called:
 - EC2_SG_2
- Open the file and modify the default value of the machine from T2.Small to T2.Micro as follows:
 - **“Default”: t2.micro**

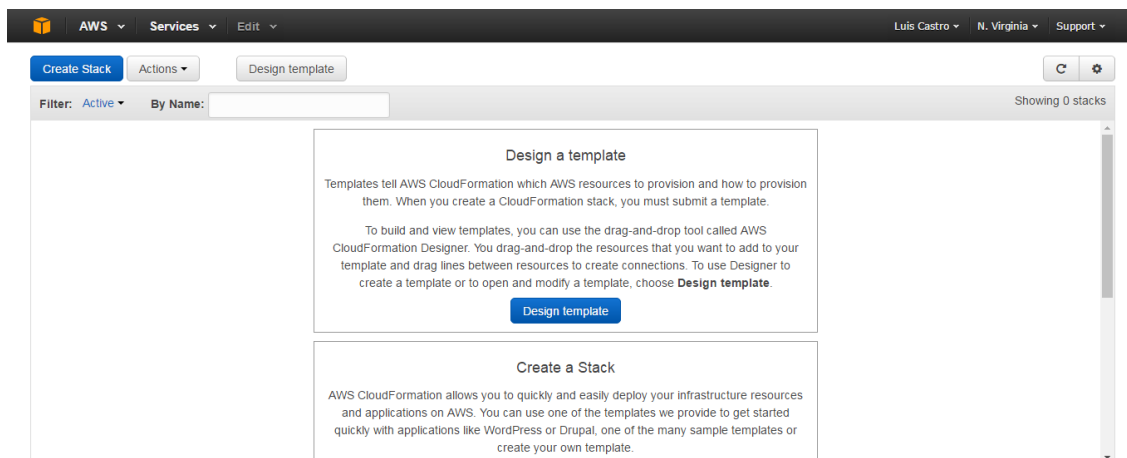
"Type": "String",

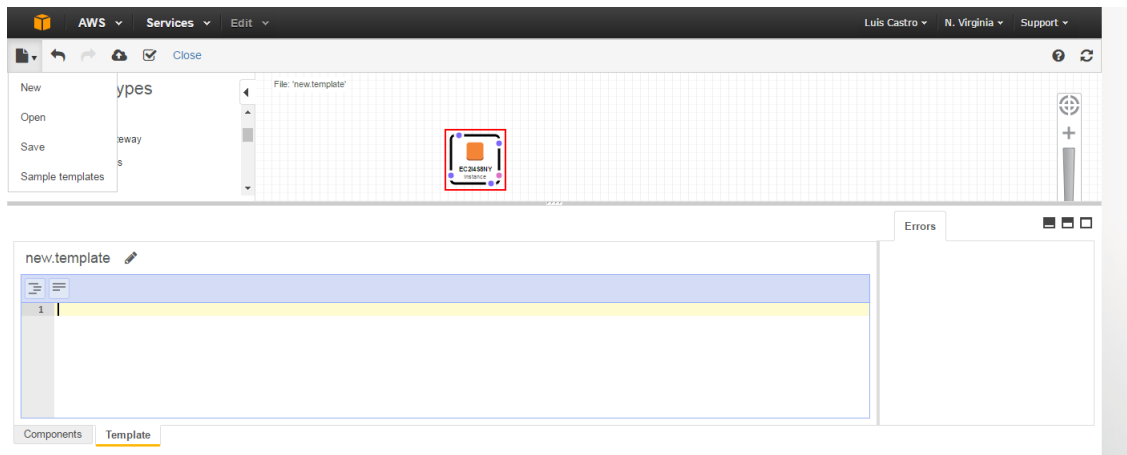
"Default": "t2.micro",

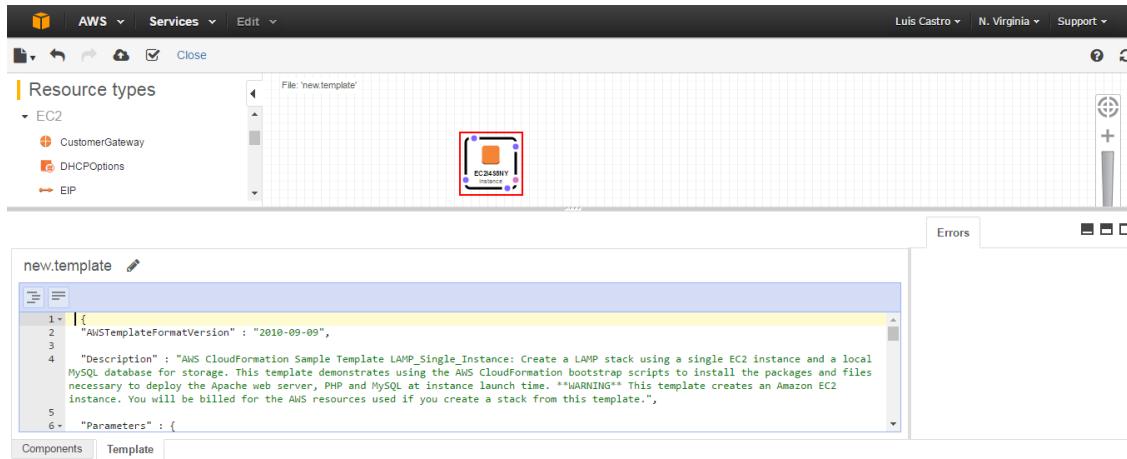
"AllowedValues": ["t1.micro", "t2.nano", "t2.micro", "t2.small", "t2.medium", "t2.large", "m1.small", "m1.medium", "m1.large", "m1.xlarge", "m2.xlarge", "m2.2xlarge", "m2.4xlarge", "m3.medium", "m3.large", "m3.xlarge", "m3.2xlarge", "m4.large", "m4.xlarge", "m4.2xlarge", "m4.4xlarge", "m4.10xlarge", "c1.medium", "c1.xlarge", "c3.large", "c3.xlarge", "c3.2xlarge", "c3.4xlarge", "c3.8xlarge", "c4.large", "c4.xlarge", "c4.2xlarge", "c4.4xlarge", "c4.8xlarge", "g2.2xlarge", "g2.8xlarge", "r3.large", "r3.xlarge", "r3.2xlarge", "r3.4xlarge", "r3.8xlarge", "i2.xlarge", "i2.2xlarge", "i2.4xlarge", "i2.8xlarge", "d2.xlarge", "d2.2xlarge", "d2.4xlarge", "d2.8xlarge", "hi1.4xlarge", "hs1.8xlarge", "cr1.8xlarge", "cc2.8xlarge", "cg1.4xlarge"]

Step 13

- Log into CloudFormation again and choose Design template
- Choose an instance again
- Delete the default template and copy the new template with the change of T2.Micro

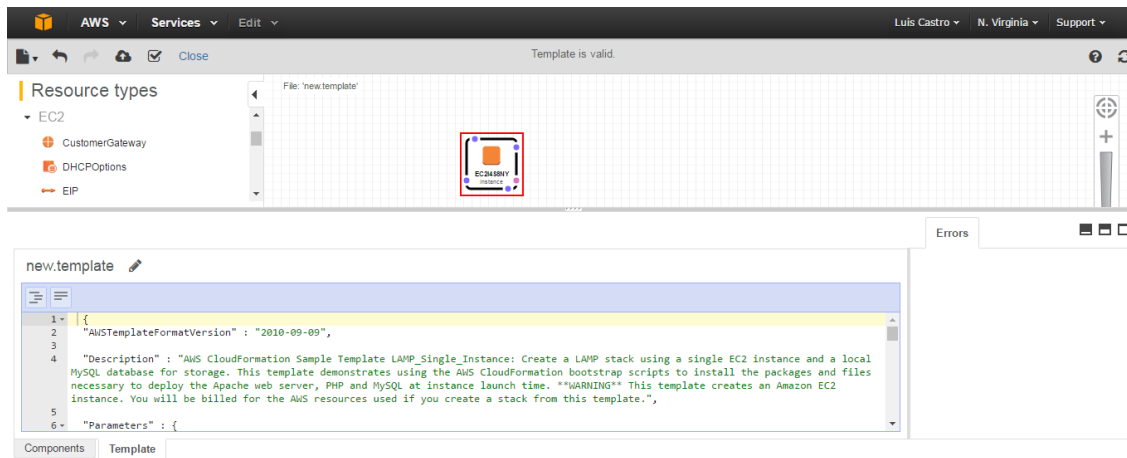




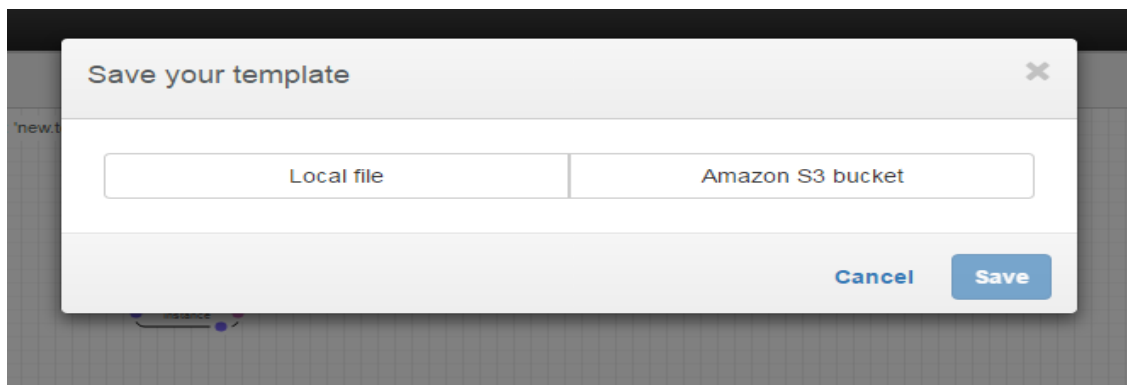


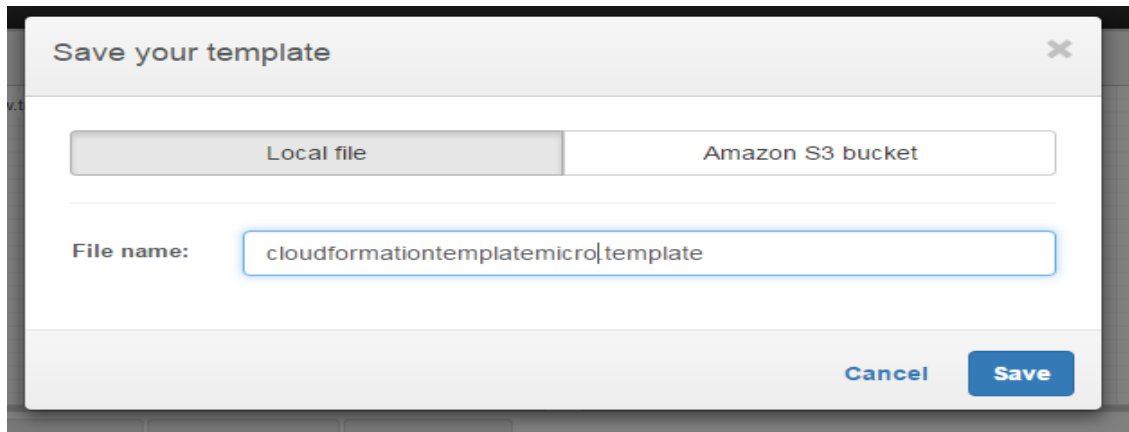
Step 14

- Validate that the template is valid



- Save template as:
 - o **Cloudformationtemplatemicro**





Save your template

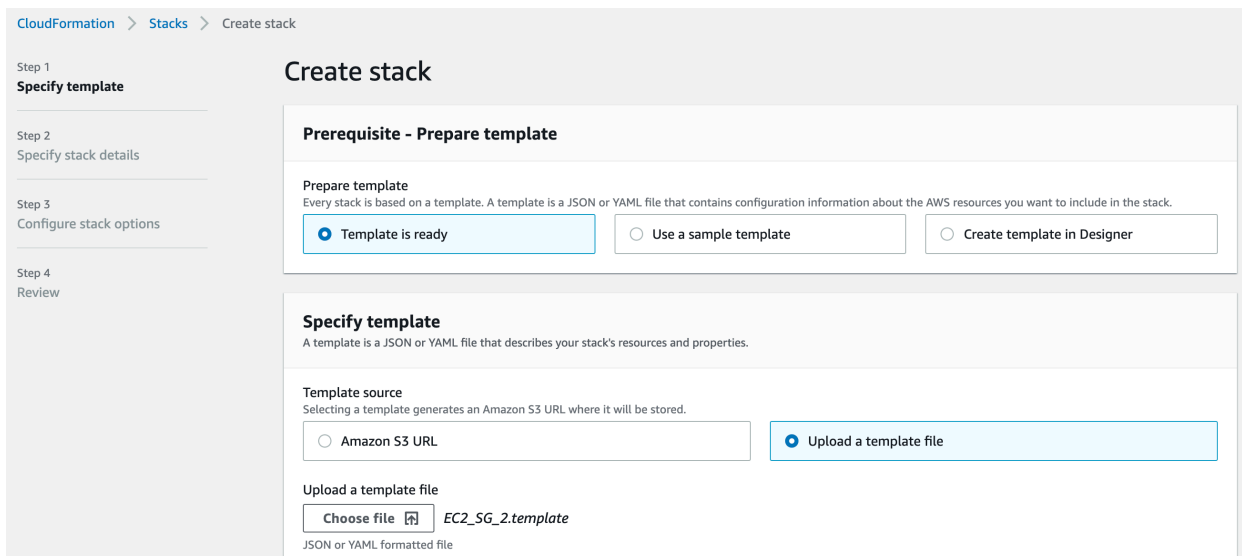
Local file Amazon S3 bucket

File name: cloudformationtemplatemicro|template

Cancel Save

Step 15

- Create a stack and select the new template
 - o Cloudformationtemplatemicro



CloudFormation > Stacks > Create stack

Step 1
Specify template

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Create 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.


☒ Template is ready ☐ Use a sample template ☐ Create 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

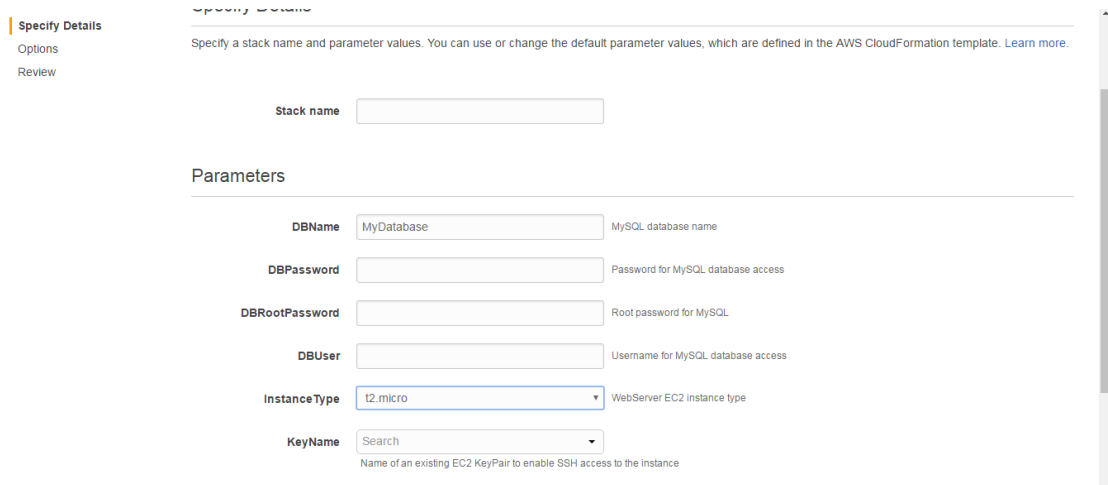
Upload a template file

Choose file  EC2_SG_2.template

JSON or YAML formatted file

Step 16

- Verify that the Instance Type parameter has the default value
 - T2.Micro



The screenshot shows the 'Specify Details' step of an AWS CloudFormation stack creation. The stack name is 'MyStack'. Under the 'Parameters' section, the following values are entered:

- DBName:** MyDatabase (MySQL database name)
- DBPassword:** (Password for MySQL database access)
- DBRootPassword:** (Root password for MySQL)
- DBUser:** (Username for MySQL database access)
- InstanceType:** t2.micro (WebServer EC2 instance type)
- KeyName:** Search (Name of an existing EC2 KeyPair to enable SSH access to the instance)

Step 17

- Terminate both created EC2s by removing the created Stacks.