

TASK: USE CLOUDFORMATION

- 1) CREATE VPC
- 2) CREATE TWO SUBNETS IN VPC
- 3) LAUNCH INSTANCES IN PUBLIC AND PRIVATE SUBNETS

AWS CLOUDFORMATION TEMPLATE:

AWSTemplateFormatVersion: 2010-09-09

Metadata:

'AWS::CloudFormation::Designer':

c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150:

size:

width: 760

height: 430

position:

x: 240

'y': 80

z: 0

embeds:

- 46ea93af-ecab-4b5c-9c50-590dbd32d140

- 588309e1-19a1-4474-88e5-3024b2a25b0e

- 83bc04c1-a66b-46e9-979b-dade36ae1cea

- 1c07859f-387e-49d4-a2f8-83d57f15a8d0

1c07859f-387e-49d4-a2f8-83d57f15a8d0:

size:

width: 260

height: 240

position:

x: 420

'y': 260

z: 1

parent: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150

embeds:

- d1f52d26-0e33-4f2c-b568-4b1801f0aa09

dependson:

- 588309e1-19a1-4474-88e5-3024b2a25b0e

83bc04c1-a66b-46e9-979b-dade36ae1cea:

size:

width: 270

height: 230

position:

x: 700

'y': 260

z: 1

parent: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150

embeds:

- 1cbac38c-74c3-40dd-a1e6-c7760fb0d8d0

46ea93af-ecab-4b5c-9c50-590dbd32d140:

size:

width: 120

height: 120

position:

x: 370

'y': 110

z: 1

parent: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150

embeds: []

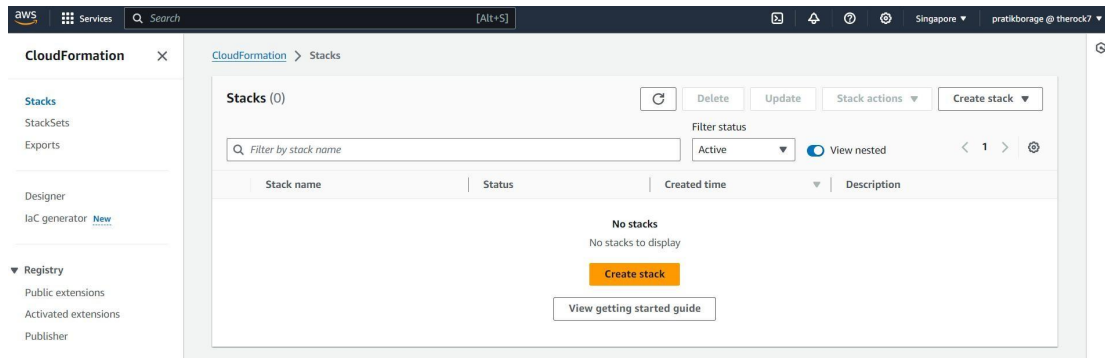
```
dependson:
  - 126f69fd-6d52-43e5-bbdb-a4db53a69c6a
d1f52d26-0e33-4f2c-b568-4b1801f0aa09:
  size:
    width: 60
    height: 60
  position:
    x: 502
    'y': 338
  z: 2
  parent: 1c07859f-387e-49d4-a2f8-83d57f15a8d0
  embeds: []
  iscontainedinside:
    - 1c07859f-387e-49d4-a2f8-83d57f15a8d0
1cbac38c-74c3-40dd-a1e6-c7760fb0d8d0:
  size:
    width: 60
    height: 60
  position:
    x: 765
    'y': 329
  z: 2
  parent: 83bc04c1-a66b-46e9-979b-dade36ae1cea
  embeds: []
  iscontainedinside:
    - 83bc04c1-a66b-46e9-979b-dade36ae1cea
    - 83bc04c1-a66b-46e9-979b-dade36ae1cea
126f69fd-6d52-43e5-bbdb-a4db53a69c6a:
  size:
    width: 60
    height: 60
  position:
    x: 60
    'y': 150
  z: 0
  embeds: []
  dependson:
    - c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150
e4415541-f052-4a57-861f-38a78e4bb978:
  source:
    id: 126f69fd-6d52-43e5-bbdb-a4db53a69c6a
  target:
    id: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150
  z: 0
588309e1-19a1-4474-88e5-3024b2a25b0e:
  size:
    width: 60
    height: 60
  position:
    x: 250
    'y': 150
  z: 1
  parent: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150
  embeds: []
  dependson:
    - 46ea93af-ecab-4b5c-9c50-590dbd32d140
Resources:
```

```
MyVPC:
  Type: 'AWS::EC2::VPC'
  Properties:
    CidrBlock: 10.0.0.0/16
  Metadata:
    'AWS::CloudFormation::Designer':
      id: c2fe8fda-2fa5-4ac2-b82a-0c897a8e1150
PublicSubnet:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref MyVPC
    AvailabilityZone: ap-southeast-1a
    CidrBlock: 10.0.0.0/24
    MapPublicIpOnLaunch: true
  Metadata:
    'AWS::CloudFormation::Designer':
      id: 1c07859f-387e-49d4-a2f8-83d57f15a8d0
  DependsOn:
    - EC2R56T9R
PrivateSubnet:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref MyVPC
    AvailabilityZone: ap-southeast-1b
    CidrBlock: 10.0.1.0/24
    MapPublicIpOnLaunch: false
  Metadata:
    'AWS::CloudFormation::Designer':
      id: 83bc04c1-a66b-46e9-979b-dade36ae1cea
PublicRouteTable:
  Type: 'AWS::EC2::RouteTable'
  Properties:
    VpcId: !Ref MyVPC
  Metadata:
    'AWS::CloudFormation::Designer':
      id: 46ea93af-ecab-4b5c-9c50-590dbd32d140
  DependsOn:
    - EC2IGZ7OO
EC2I3H316:
  Type: 'AWS::EC2::Instance'
  Properties:
    ImageId: ami-0702b1aec03c6a3a4
    InstanceType: t2.micro
    KeyName: mongokey
  Metadata:
    'AWS::CloudFormation::Designer':
      id: d1f52d26-0e33-4f2c-b568-4b1801f0aa09
EC2IKJY7:
  Type: 'AWS::EC2::Instance'
  Properties:
    ImageId: ami-0702b1aec03c6a3a4
    InstanceType: t2.micro
    KeyName: mongokey
  Metadata:
    'AWS::CloudFormation::Designer':
      id: 1cbac38c-74c3-40dd-a1e6-c7760fb0d8d0
EC2IGZ7OO:
```

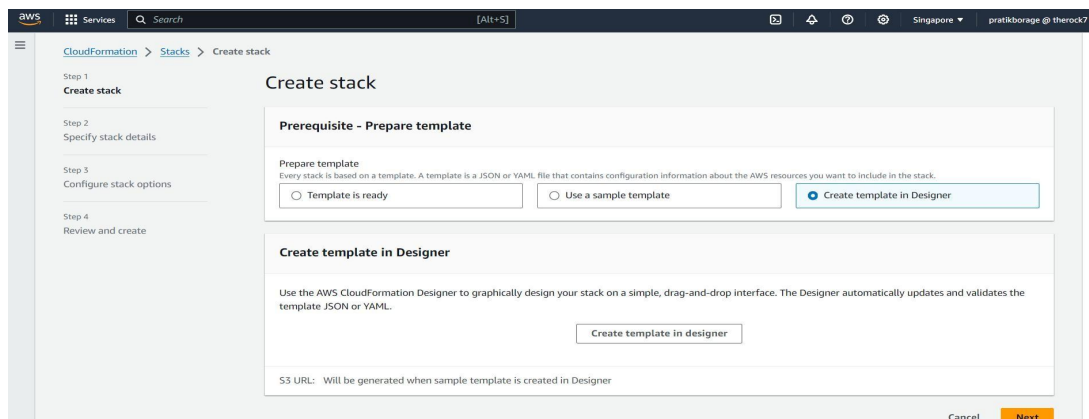
```

Type: 'AWS::EC2::InternetGateway'
Properties: {}
Metadata:
  'AWS::CloudFormation::Designer':
    id: 126f69fd-6d52-43e5-bbdb-a4db53a69c6a
DependsOn:
  - MyVPC
EC2VPCGSGZY:
Type: 'AWS::EC2::VPCGatewayAttachment'
Properties:
  InternetGatewayId: !Ref EC2IGZ7OO
  VpcId: !Ref MyVPC
Metadata:
  'AWS::CloudFormation::Designer':
    id: e4415541-f052-4a57-861f-38a78e4bb978
EC2R56T9R:
Type: 'AWS::EC2::Route'
Properties:
  DestinationCidrBlock: 0.0.0.0/0
  RouteTableId: !Ref PublicRouteTable
  GatewayId: !Ref EC2IGZ7OO
Metadata:
  'AWS::CloudFormation::Designer':
    id: 588309e1-19a1-4474-88e5-3024b2a25b0e
DependsOn:
  - PublicRouteTable
  
```

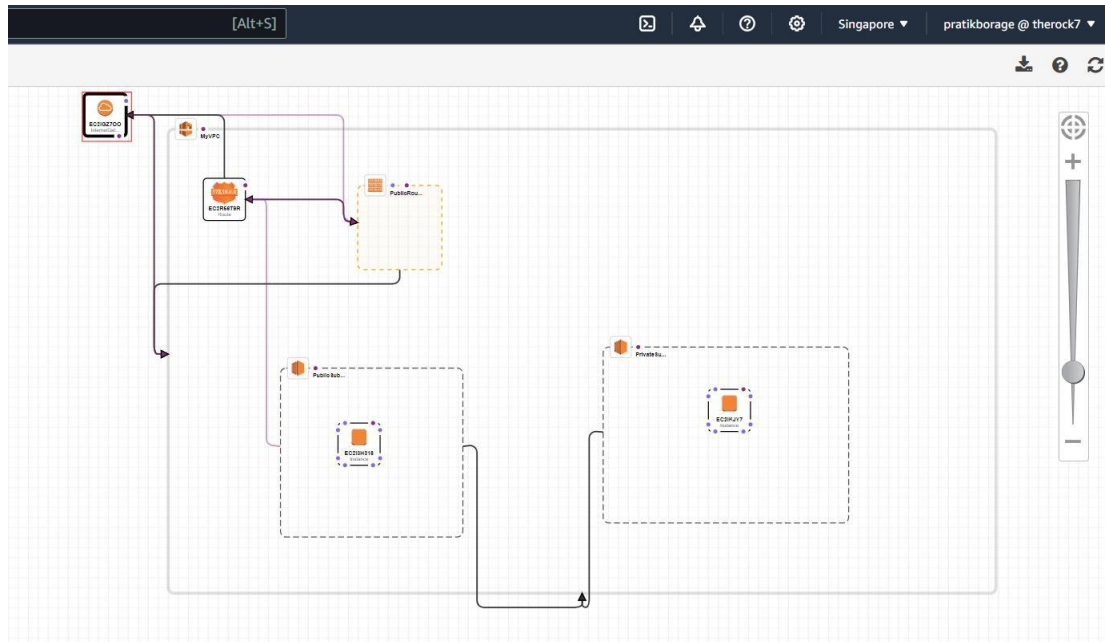
STEP 1:- Go to Cloudformation -->Create Stack



STEP2:- Create Template in Designer



STEP 3:- Drag and drop resources
Make the required connections



Validate the template
And **create stack**
Click on **NEXT**

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 ☐ Sync from Git - new

Provide an Amazon S3 URL to your template. Upload your template directly to the console. Sync a template from your Git repository.

Amazon S3 URL
<https://s3-ap-southeast-1.amazonaws.com/cf-templates-1hh45xingkc1w-ap-southeast-1/20240434U2-new.template.tgzpgmxxq>

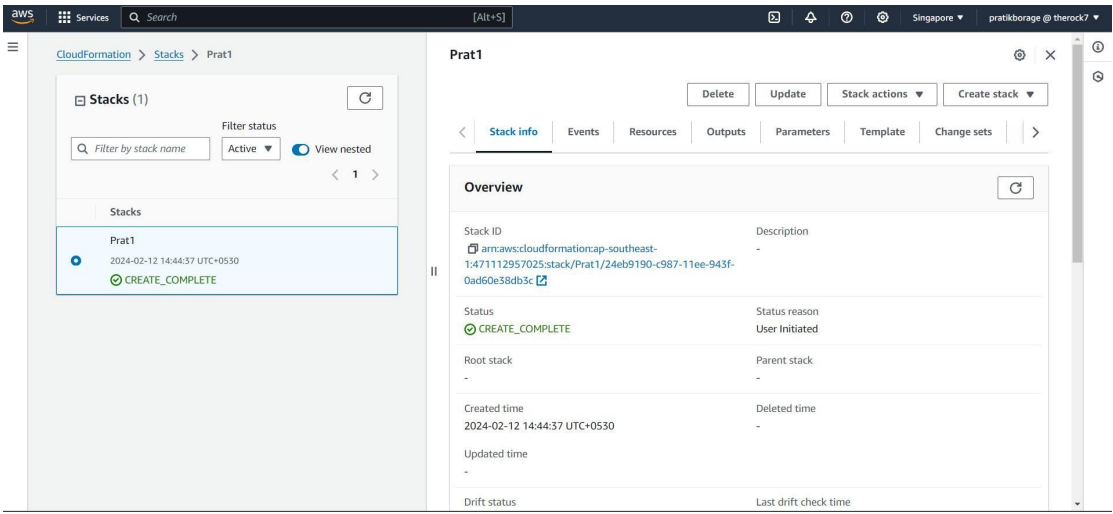
Amazon S3 template URL
S3 URL: <https://s3-ap-southeast-1.amazonaws.com/cf-templates-1hh45xingkc1w-ap-southeast-1/20240434U2-new.template.tgzpgmxxq>

[View in Designer](#) [Cancel](#) [Next](#)

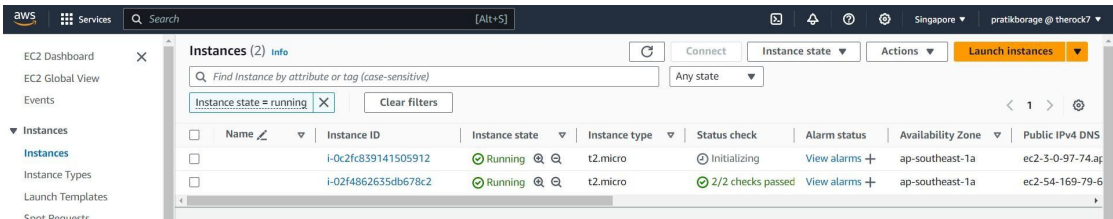
STEP 4: Review the page

-->Submit

Stack creation is started



Launched Instances:-



Subnets Created:-

