

Karthik Pradeep Hegadi

2KE20CS032

### Assignment 47

---

*Understood. To follow the provided instructions and create the files/directory using the same name and case as provided in the task steps, please provide me with the specific names and case instructions for the files/directory you want to create.*

---

---

## AWS

---

---

### Assignment: 1 : Auto Deployment using Cloudformation Template

---

**Note\*\* initially I had aws image in which I have installed nginx I used it**

---

#### Cloud formation template in yaml

1.Open the delta-vpc-cfn.yaml that is attached in this link

[https://drive.google.com/file/d/1id50jMv1Md1Ctxk2hNNdfMrfaQVk8W76/view?](https://drive.google.com/file/d/1id50jMv1Md1Ctxk2hNNdfMrfaQVk8W76/view?usp=)

usp=

sharing

2.Go through the file and understand the code

3. You need to couple of resources in the yaml file for creating security group and instance of Type: AWS::EC2::SecurityGroup & Type: AWS::EC2::Instance

4.You need to use Nginx ami image id for instance creation.

5. you need to allow port 80 as inbound and outbound, port 22 as inbound

6.Once you added these resources you can save the file

7. You can use this AWS link for your reference to add the resources

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-property-es-ec2-security-group.html>

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-property-es-ec2-instance.html>

```

SecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupName: !Sub ${EnvironmentName}-SecurityGroup
    GroupDescription: Security Group for Nginx EC2 instances
    VpcId: !Ref VPC
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0

NginxInstance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: t2.micro
    ImageId: ami-089131dd76c790637 # Replace with the actual AMI ID for Nginx my ami id
    KeyName: key key to access
    SecurityGroupIds:
      - !Ref SecurityGroup
    SubnetId: !Ref PublicSubnet1 # Choose the appropriate subnet for your instance

```

8. Navigate to Launch templates(Ec2 feature)
9. Navigate to Cloud formation page
10. Click on create stack
11. In the Template source select upload a template file
12. Choose the modified codebuild-vpc-cfn.yaml click on "Next"
13. Provide the stack name
14. In the Parameters you can change EnvironmentName to your desired name
15. Click 'Next'
16. You can add the tags, leave other options as default and click "Next"
17. Review your stack details once and you can click on create stack
18. You can see the status of the stack when it started creating

# Specify stack details

### Provide a stack name

Stack name

nginx-deployment-cloudformation

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.

#### EnvironmentName

An environment name that is prefixed to resource names

Marak-CFN

#### PrivateSubnet1CIDR

Please enter the IP range (CIDR notation) for the private subnet in the first Availability Zone

10.10.101.0/24

#### PrivateSubnet2CIDR

Please enter the IP range (CIDR notation) for the private subnet in the second Availability Zone

10.10.201.0/24

#### PublicSubnet1CIDR

Please enter the IP range (CIDR notation) for the public subnet in the first Availability Zone

10.10.1.0/24

outh-  
dformation/141a83c0-

ment-cloudformation

Filter status

Active

< 1 >

## nginx-deployment-cloudformation

Delete Update Stack actions Create stack

Stack info Events Resources Outputs Parameters Template Change sets Git sync - new

### Events (41)

Detect root cause

Search events

Timestamp	Logical ID	Status	Status reason
2023-12-30 20:52:32 UTC+0530	nginx-deployment-cloudformation	CREATE_COMPLETE	-
2023-12-30 20:52:31 UTC+0530	NginxInstance	CREATE_COMPLETE	-
2023-12-30 20:52:06 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_COMPLETE	-
2023-12-30 20:52:06 UTC+0530	PublicSubnet2RouteTableAssociation	CREATE_COMPLETE	-
2023-12-30 20:52:06 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_IN_PROGRESS	Resource creation Initiated
2023-12-30 20:52:06 UTC+0530	PublicSubnet2RouteTableAssociation	CREATE_IN_PROGRESS	Resource creation Initiated
2023-12-30 20:52:06 UTC+0530	DefaultPublicRoute	CREATE_COMPLETE	-
2023-12-30 20:52:05 UTC+0530	DefaultPublicRoute	CREATE_IN_PROGRESS	Resource creation Initiated
2023-12-30 20:52:04 UTC+0530	PublicSubnet1RouteTableAssociation	CREATE_IN_PROGRESS	-
2023-12-30 20:52:04 UTC+0530	PublicSubnet2RouteTableAssociation	CREATE_IN_PROGRESS	-
2023-12-30 20:52:04 UTC+0530	DefaultPublicRoute	CREATE_IN_PROGRESS	-

### Instances (1) Info

Find Instance by attribute or tag (case-sensitive)

running

Clear filters

Connect Instance state Actions Launch

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Avai
	i-00ea407b36270699f	Running	t2.micro	2/2 checks passed	No alarms	ap-s

EC2 > Instances > i-00ea407b36270699f

Instance summary for i-00ea407b36270699f [Info](#)

Updated less than a minute ago

Instance ID

i-00ea407b36270699f

IPv6 address

-

Hostname type

IP name: ip-10-10-1-231.ap-south-1.compute.internal

Answer private resource DNS name

Public IPv4 address

3.110.29.22 [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-10-10-1-231.ap-south-1.compute.internal

Instance type

Private IPv4 addresses

10.10.1.231

Public IPv4 DNS

ec2-3-110-29-22.ap-south-1.compute.amazonaws.com [address](#)

Elastic IP addresses

b-1+job X

Key pairs | EC2 | ap- X

Instance details | EC X

Welcome to nginx! X

CloudFormation - St X

Instance details | EC X

Server Not Found X

3.110.29.22

# Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](#).  
Commercial support is available at [nginx.com](#).

*Thank you for using nginx.*