

Learning Outcomes

- Using the CloudFormation CLI
- Observe the JSON structure used in CloudFormation
- Observe the process of stack creation in CloudFormation

How to do it

- Download the file *CF1.json* and *CF2.json* provided along with this DIY
- Open the AWS Console and navigate to the VPC section
- Note down the ID of the default VPC already created

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Main Route table
vpc-daa3c5a0	available	172.31.0.0/16	-	dopt-7724270c	rtb-4d2ac733	

VPC: vpc-daa3c5a0	
Description	CIDR Blocks
VPC ID: vpc-daa3c5a0 State: available IPv4 CIDR: 172.31.0.0/16 IPv6 CIDR: - Network ACL: acl-c817deb5 DHCP options set: dopt-7724270c Route table: rtb-4d2ac733	Tenancy: default Default VPC: Yes Classic Link: Disabled DNS resolution: Enabled DNS hostnames: Enabled ClassicLink DNS Support: Disabled Owner: 511948551519

- Navigate to the subnet section by clicking on “Subnets” on the left side.
- Note down the subnet ID of the default subnet already created for the default VPC

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone
	subnet-0ca9a503	available	vpc-daa3c5a0	172.31.64.0/20	4091	-	us-east-1f

Subnet: subnet-0ca9a503

Description	Flow Logs	Route Table	Network ACL	Tags	Sharing
Subnet ID: subnet-0ca9a503 VPC: vpc-daa3c5a0 Available IPv4 Addresses: 4091 Availability Zone: us-east-1f (use1-az5) Network ACL: acl-c817deb5					
State: available IPv4 CIDR: 172.31.64.0/20 IPv6 CIDR: - Route Table: rtb-4d2ac733 Default subnet: Yes					

- Click on “Create Subnet” at the top of the screen
- Fill in the form as shown below and click on Create

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

The screenshot shows the AWS VPC console interface for creating a new subnet. The URL in the browser is `console.aws.amazon.com/vpc/home?region=us-east-1#CreateSubnet:SubnetId=subnet-0ca9a503`. The page title is "Create subnet". The form fields are as follows:

- Name tag: pb_subnet
- VPC*: vpc-daa3c5a0
- VPC CIDRs:

CIDR	Status	Status Reason
172.31.0.0/16	associated	
- Availability Zone: us-east-1f
- IPv4 CIDR block*: 172.31.56.0/22

At the bottom, there is a note: "* Required" and two buttons: "Cancel" and "Create".

Make sure the IPv4 CIDR block is not being used by any subnet already created, and that the availability zone is a different one than that used by the default subnet.

- Note down the Subnet ID of the subnet just created.

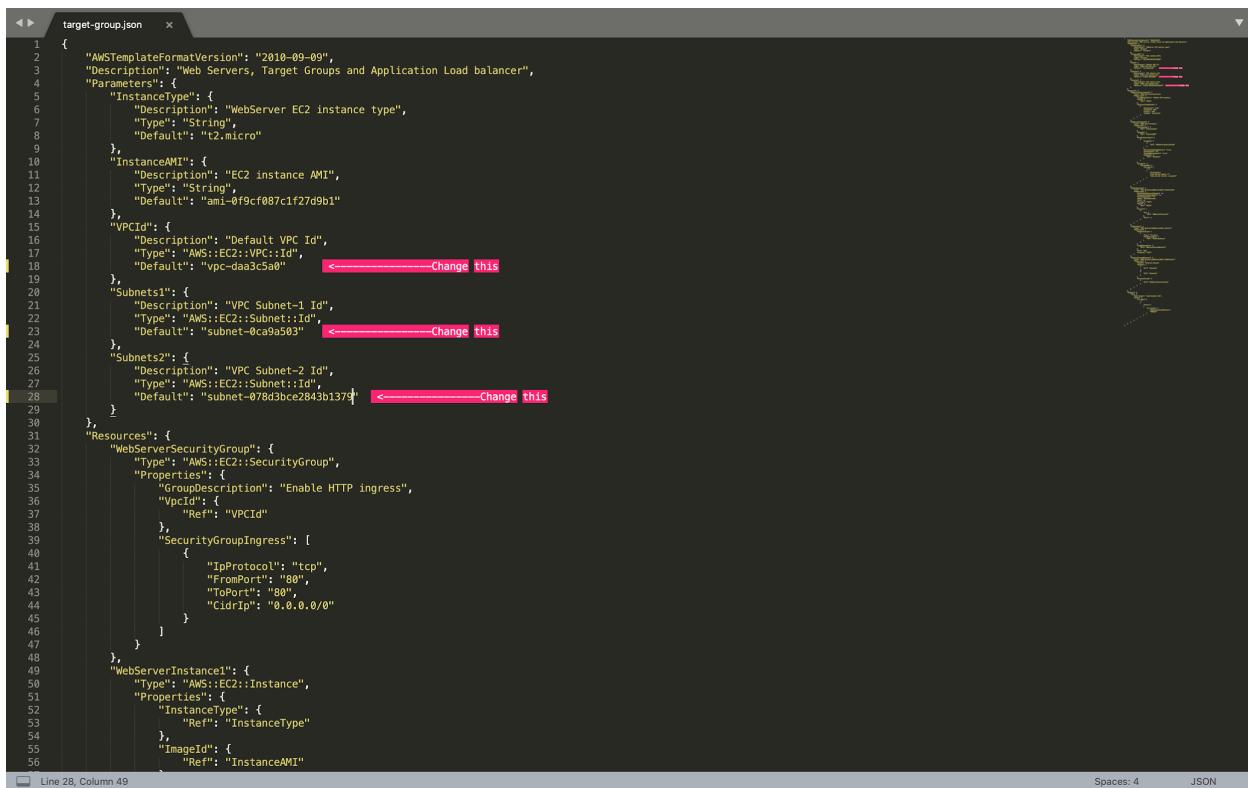
The screenshot shows the AWS VPC Management Subnets page. On the left, there's a sidebar with various VPC-related options like Virtual Private Cloud, Your VPCs, Subnets, Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, Endpoint Services, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups. The 'Subnets' option is selected. The main area displays a table of subnets with columns: Name, Subnet ID, State, VPC, IPv4 CIDR, Available IPv4, IPv6 CIDR, and Availability Zone. Two subnets are listed: 'pb_subnet' (subnet-078d3bce2843b1379) and 'subnet-0ca9a503'. Below the table, a detailed view for 'pb_subnet' is shown with tabs for Description, Flow Logs, Route Table, Network ACL, Tags, and Sharing. The 'Description' tab is active, showing details like Subnet ID, VPC, Available IPv4 Addresses, Availability Zone, Network ACL, State, IPv4 CIDR, IPv6 CIDR, Route Table, and Default subnet.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone
pb_subnet	subnet-078d3bce2843b1379	available	vpc-daa3c5a0	172.31.56.0/22	1019	-	us-east-1f
	subnet-0ca9a503	available	vpc-daa3c5a0	172.31.64.0/20	4091	-	us-east-1f

- Open the CF1.json file in your preferred text editor
- Change the values marked below in lines 18,23 and 28, with the values of the default VPC ID, default subnet ID and the ID of the created subnet respectively
- Make the above changes in lines 18,23 and 28 for the file CF2.json as well after opening it in a text editor.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.



```

1 {
2     "AWSTemplateFormatVersion": "2010-09-09",
3     "Description": "Web Servers, Target Groups and Application Load balancer",
4     "Parameters": {
5         "InstanceType": {
6             "Description": "WebServer EC2 instance type",
7             "Type": "String",
8             "Default": "t2.micro"
9         },
10        "InstanceAMI": {
11            "Description": "EC2 instance AMI",
12            "Type": "String",
13            "Default": "ami-0f9cf087c1f27d9b1"
14        },
15        "VPCId": {
16            "Description": "Default VPC Id",
17            "Type": "AWS::EC2::VPC::Id",
18            "Default": "vpc-daa3c5a0" Change this
19        },
20        "Subnets1": {
21            "Description": "VPC Subnet-1 Id",
22            "Type": "AWS::EC2::Subnet::Id",
23            "Default": "subnet-0ca9a503" Change this
24        },
25        "Subnets2": {
26            "Description": "VPC Subnet-2 Id",
27            "Type": "AWS::EC2::Subnet::Id",
28            "Default": "subnet-078d3bce2843b137" Change this
29    }
30 },
31 "Resources": {
32     "WebServerSecurityGroup": {
33         "Type": "AWS::EC2::SecurityGroup",
34         "Properties": {
35             "GroupDescription": "Enable HTTP ingress",
36             "VpcId": {
37                 "Ref": "VPCId"
38             },
39             "SecurityGroupIngress": [
40                 {
41                     "IpProtocol": "tcp",
42                     "FromPort": "80",
43                     "ToPort": "80",
44                     "CidrIp": "0.0.0.0/0"
45                 }
46             ]
47         }
48     },
49     "WebServerInstance1": {
50         "Type": "AWS::EC2::Instance",
51         "Properties": {
52             "InstanceType": {
53                 "Ref": "InstanceType"
54             },
55             "ImageId": {
56                 "Ref": "InstanceAMI"
57             }
58         }
59     }
60 }
61 
```

Line 28, Column 49 Spaces: 4 JSON

- Save and close the file
- Open your terminal (assuming AWS CLI is already installed and configured)
- Navigate to the folder where the JSONs file are stored
- Enter the following command and press Enter
`aws cloudformation create-stack --stack-name gltest-stack --template-body file://CF1.json`

Whoops! You get the following error

An error occurred (ValidationError) when calling the CreateStack operation: Template format error: Unresolved resource dependencies [NoSecGroup] in the Resources block of the template

Let's open the file and navigate to line 183. As you can see, the security group for the Load Balancer has been set to "NoSecGroup". However, there is no security group as such created in the JSON file. If you navigate to line 32, you will see that the name of the security group created is "WebServerSecurityGroup".

- Go back to the terminal and type the command
`aws cloudformation create-stack --stack-name gltest-stack --template-body file://CF2.json`

If you open this file and check, you can see that the security group for the load balancer has been correctly set.

- Go back to the AWS console and navigate to CloudFormation

The screenshot shows the AWS CloudFormation Stacks page. At the top, there are buttons for Create, Delete, Update, Stack actions, and a prominent orange Create stack button. Below this is a search bar labeled 'Filter by stack name' and a dropdown for 'Active'. A 'View nested' checkbox is also present. The main table lists one stack:

Stack name	Status	Created time	Description
glttest-stack	CREATE_IN_PROGRESS	2019-09-28 15:50:07 UTC+0530	Web Servers, Target Groups and Application Load balancer

- Click on the stack and follow the steps of its creation. Wait for a few minutes for stack

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

creation to finish.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

The screenshot shows the AWS CloudFormation console with the URL console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/stackinfo?filteringText=&filteringStatus=active&viewNested=.... The page displays the 'gltest-stack' details. The stack is currently in the 'CREATE_IN_PROGRESS' state, created on 2019-09-28 at 15:50:07 UTC+0530. The status reason is '-'.

Stack ID	Description
arn:aws:cloudformation:us-east-1:511948551519:stack/gltest-stack/8baaa500-e1d9-11e9-b154-0a182a3cd028	Web Servers, Target Groups and Application Load balancer

Overview

Status	Status reason
① CREATE_IN_PROGRESS	-

Root stack	Parent stack
-	-

Created time	Deleted time
2019-09-28 15:50:07 UTC+0530	-

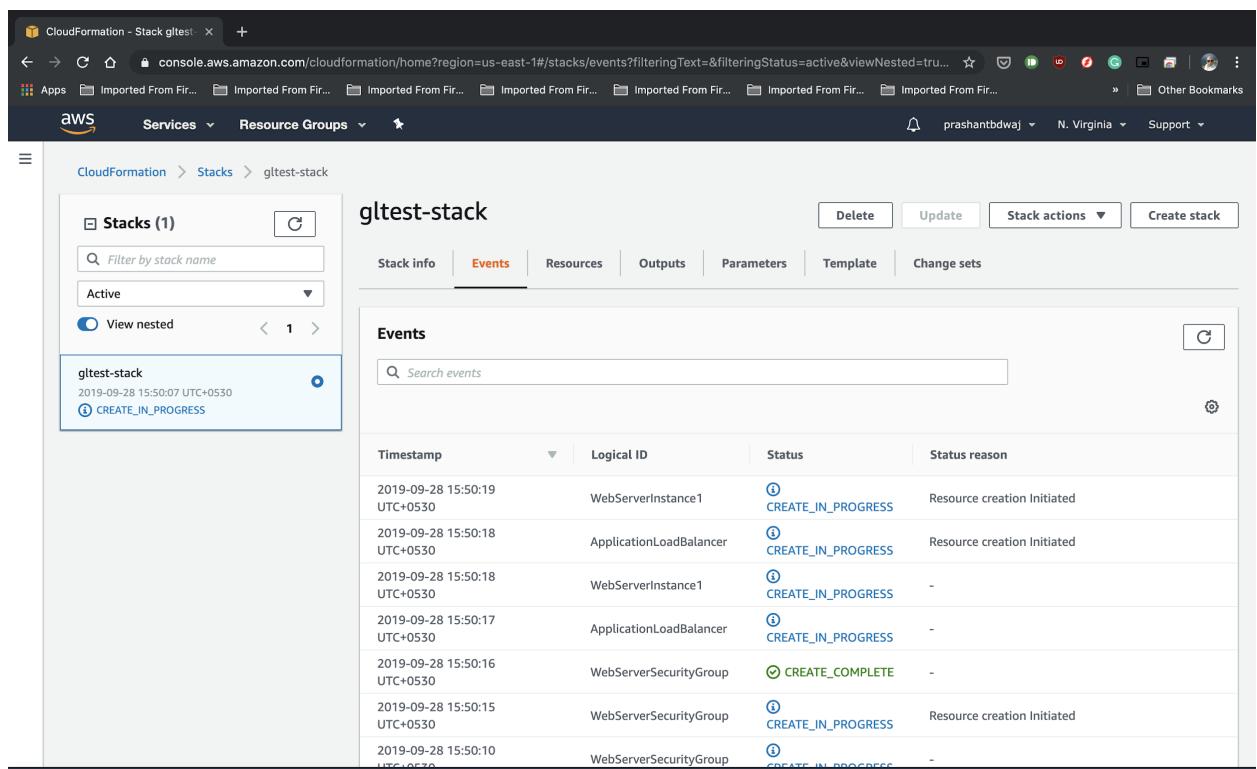
Updated time	-
-	-

Drift status	Last drift check time
⊖ NOT_CHECKED	-

Termination protection	IAM role
Disabled	-

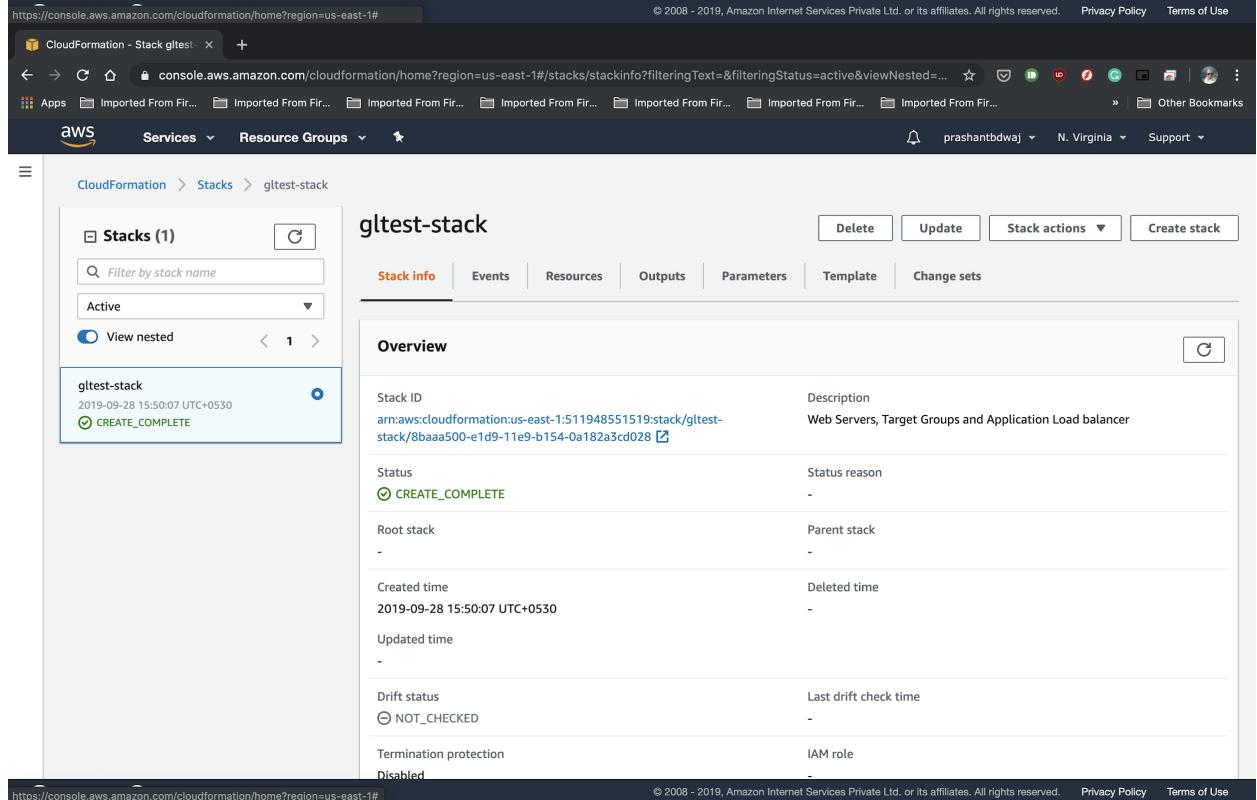
This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.



The screenshot shows the AWS CloudFormation console with the 'Events' tab selected for the 'gltest-stack'. The table lists several events from September 28, 2019, at 15:50 UTC+0530, indicating the creation of various resources like WebServerInstance1, ApplicationLoadBalancer, and WebServerSecurityGroup.

Timestamp	Logical ID	Status	Status reason
2019-09-28 15:50:19 UTC+0530	WebServerInstance1	CREATE_IN_PROGRESS	Resource creation Initiated
2019-09-28 15:50:18 UTC+0530	ApplicationLoadBalancer	CREATE_IN_PROGRESS	Resource creation Initiated
2019-09-28 15:50:18 UTC+0530	WebServerInstance1	CREATE_IN_PROGRESS	-
2019-09-28 15:50:17 UTC+0530	ApplicationLoadBalancer	CREATE_IN_PROGRESS	-
2019-09-28 15:50:16 UTC+0530	WebServerSecurityGroup	CREATE_COMPLETE	-
2019-09-28 15:50:15 UTC+0530	WebServerSecurityGroup	CREATE_IN_PROGRESS	Resource creation Initiated
2019-09-28 15:50:10 UTC+0530	WebServerSecurityGroup	CREATE_IN_PROGRESS	-



The screenshot shows the AWS CloudFormation console with the 'Stack info' tab selected for the 'gltest-stack'. The table provides detailed information about the stack's creation, including its ID, status, and creation time.

Stack ID	arn:aws:cloudformation:us-east-1:511948551519:stack/gltest-stack/8baaa500-e1d9-11e9-b154-0a182a3cd028	Description	Web Servers, Target Groups and Application Load balancer
Status	CREATE_COMPLETE	Status reason	-
Root stack	-	Parent stack	-
Created time	2019-09-28 15:50:07 UTC+0530	Deleted time	-
Updated time	-		
Drift status	NOT_CHECKED	Last drift check time	-
Termination protection	Disabled	IAM role	-

- When stack creation is completed, navigate to the Outputs tab.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

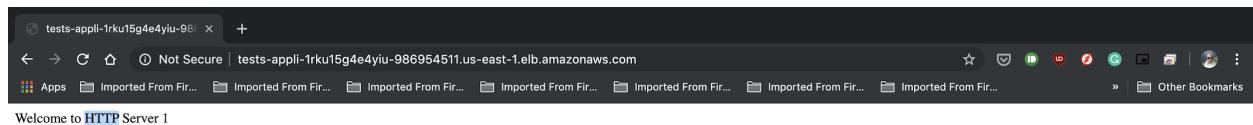
The screenshot shows the AWS CloudFormation console with the 'gltest-stack' details. The 'Outputs' tab is selected, showing the following output:

Key	Value	Description	Export name
URL	http://gltes-appli-10BUJTJ6F4IO-16605831.us-east-1.elb.amazonaws.com	Load Balancer URL	-

- Click on the URL value to confirm that Apache was installed and hence the load balancer was created successfully using the CloudFormation Template.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.
Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited



- Navigate back to the CloudFormation console, select the stack and click on Delete to delete the stack.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited

The screenshot shows the AWS CloudFormation console interface. On the left, there's a sidebar with navigation links: 'CloudFormation' (selected), 'Stacks' (highlighted in orange), 'Stack details', 'Drifts', 'StackSets', 'Exports', and 'Designer'. Below the sidebar are links for 'Previous console' and 'Feedback'. The main content area is titled 'CloudFormation > Stacks' and shows a table titled 'Stacks (1)'. The table has columns: Stack name, Status, Created time, and Description. One row is visible: 'gltest-stack' with status 'CREATE_COMPLETE', created on '2019-09-28 15:50:07 UTC+0530', and a description 'Web Servers, Target Groups an...'. At the top of the main area, there are buttons for 'Create stack', 'Delete', 'Update', 'Stack actions', and a dropdown for 'Active'. There's also a 'View nested' button. The bottom of the page includes standard footer links: 'Feedback', 'English (US)', '© 2008 - 2019, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.', 'Privacy Policy', and 'Terms of Use'.

This file is meant for personal use by demay.tom@gmail.com only.

Sharing or publishing the contents in part or full is liable for legal action.

Proprietary content. ©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited