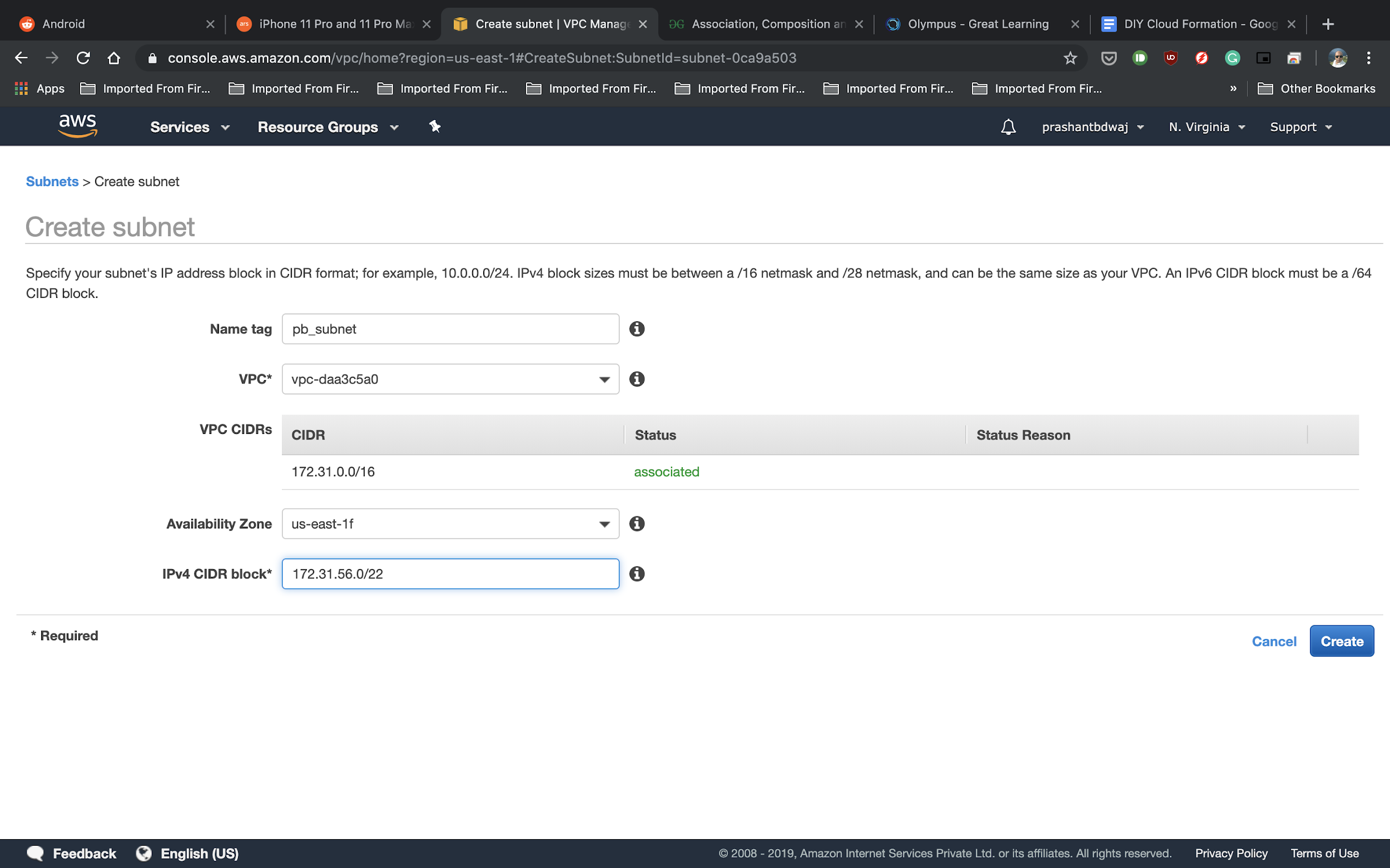


#### Learning Outcomes

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

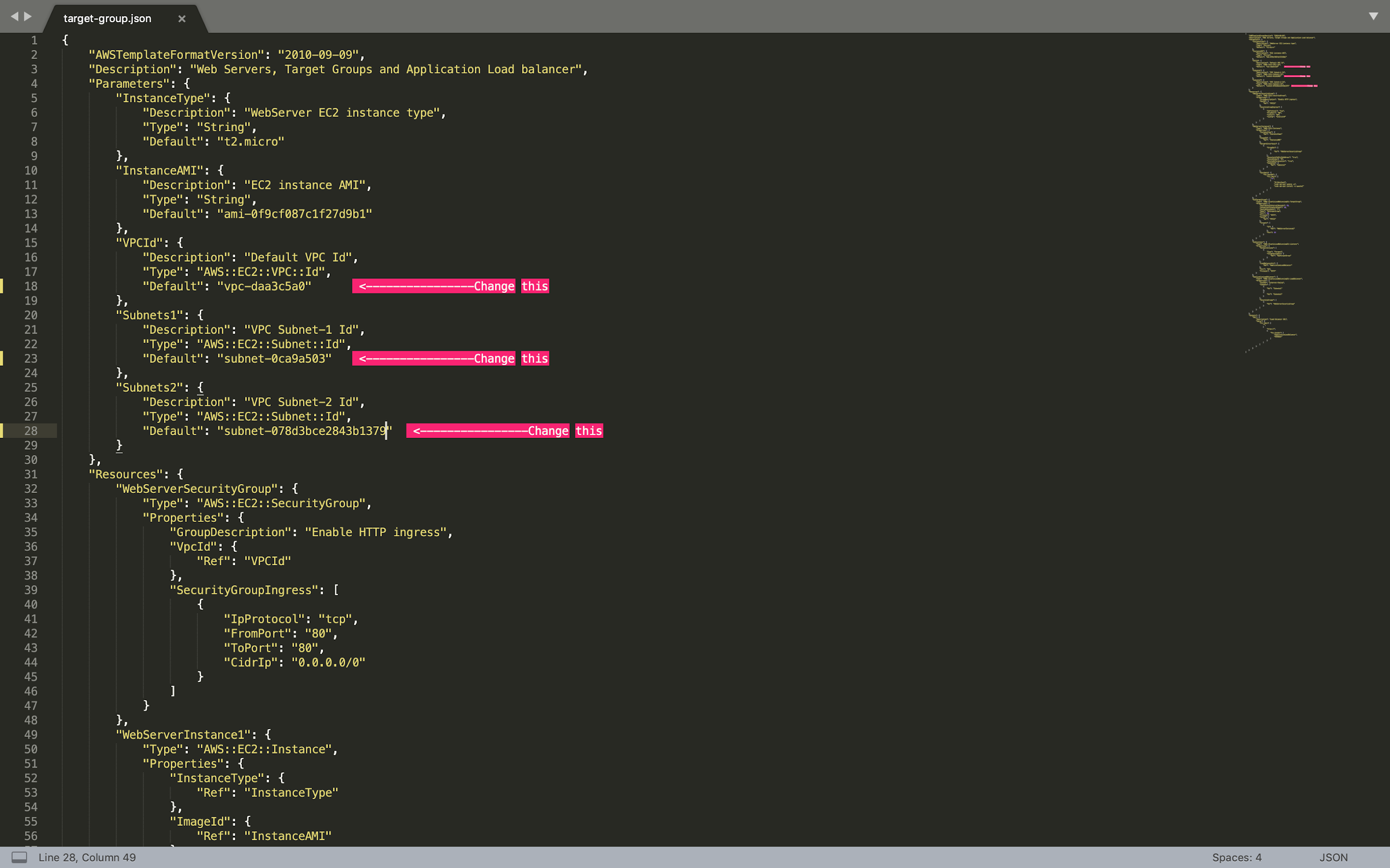
#### How to do it.

* 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
* 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
* Click on “Create Subnet” at the top of the screen
* Fill in the form as shown below and click on 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.
* 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
* Repeat the above step for the file CF2.json as well i.e. open CF2.json and replace the appropriate values on lines 18,23, and 28.



* 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 should encounter 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
* Click on the stack and follow the steps of its creation. Wait for a few minutes for stack creation to finish.
* When stack creation is completed, navigate to the Outputs tab.
* Click on the URL value to confirm that Apache was installed and hence the load balancer was created successfully using the CloudFormation Template.
* Navigate back to the CloudFormation console, select the stack and click on Delete to delete the stack.

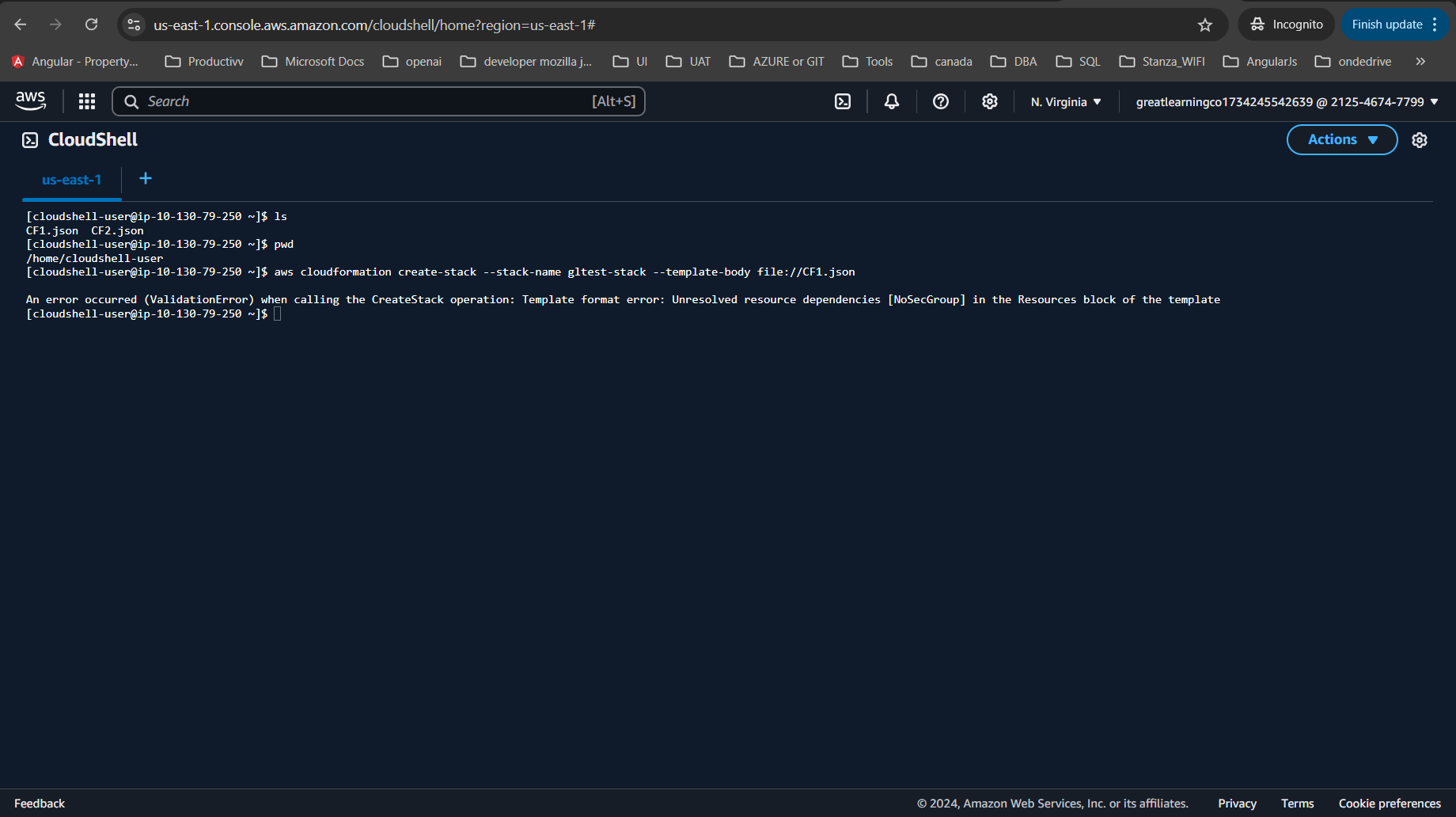
Marks Distribution

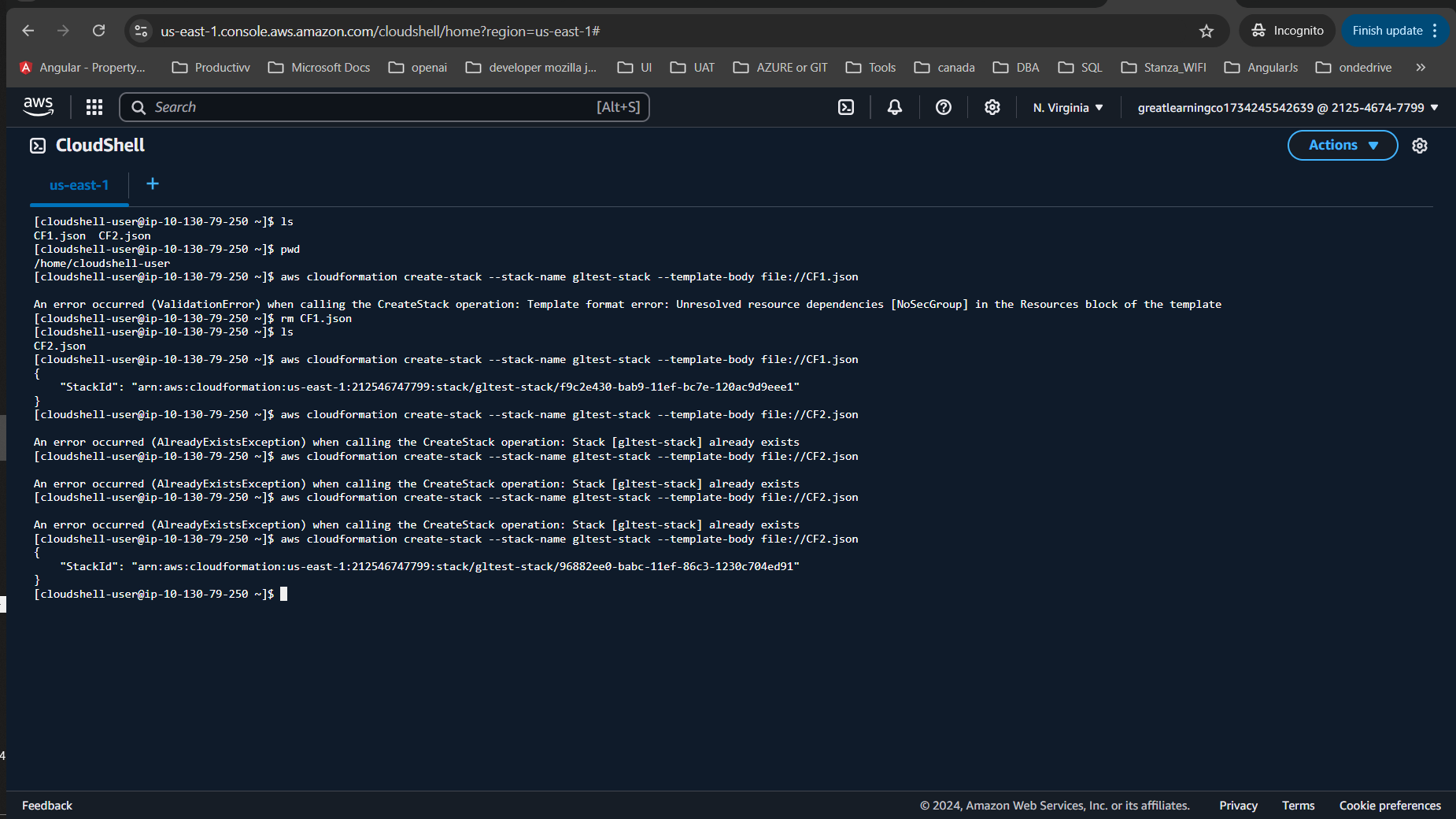
1. Log of stack creation using CF1.json 5 marks
2. Error message when using CF1.json 5 marks
3. Log of stack creation using CF2.json 5 marks
4. URL in Web page 5 marks

A screenshot of a computer

Description automatically generated

Error





A screenshot of a computer

Description automatically generated

After remove nosecgroup

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Steps to be followed

**1. Download the Files**

Ensure you’ve downloaded the **CF1.json** and **CF2.json** files provided with your DIY. These files will contain the CloudFormation template definitions you’ll work with.

**2. Set Up Your AWS Environment**

1. **Open the AWS Console** and **navigate to the VPC section**:
   * In the AWS Console, go to the **VPC** service from the navigation menu or search for "VPC".
   * Note down the **ID of the default VPC** created by AWS in your account. This VPC ID will be used in the CloudFormation templates.
2. **Navigate to the Subnet section**:
   * In the VPC dashboard, on the left sidebar, click on **Subnets**.
   * Note the **Subnet ID** of the default subnet created for the default VPC.

**3. Create a New Subnet**

1. **Create a New Subnet**:
   * Click on **Create Subnet** at the top of the screen.
   * Fill in the form with the following parameters (ensure the CIDR block does not conflict with any existing subnet and that it is in a different Availability Zone):
     + **VPC ID**: Choose the default VPC (the one noted in the previous step).
     + **CIDR Block**: Choose a subnet CIDR block that’s not in use (e.g., 10.0.2.0/24).
     + **Availability Zone**: Choose a different Availability Zone than the default subnet.
2. **Note down the Subnet ID**:
   * After creating the subnet, note the **Subnet ID** of the newly created subnet. You will need this for your CloudFormation templates.

**4. Edit the CloudFormation Templates (CF1.json & CF2.json)**

1. **Open the CF1.json file in a text editor**:
   * Find the following lines and update them:
     + **Line 18**: Replace the default VPC ID (which you noted earlier).
     + **Line 23**: Replace the default subnet ID (the one you noted earlier).
     + **Line 28**: Replace it with the ID of the newly created subnet.
2. **Repeat the above steps for CF2.json**:
   * Open CF2.json and make the same changes to lines 18, 23, and 28, replacing them with the correct IDs.
3. **Save the changes** and close the files.

**5. Create the CloudFormation Stack Using CF1.json**

1. **Open your terminal** (ensure AWS CLI is installed and configured with your AWS credentials).
2. **Navigate to the folder where your JSON files are stored**:
   * Use cd <folder-path> to navigate to the directory containing CF1.json and CF2.json.
3. **Run the following command** to create the stack using CF1.json:

bash

Copy code

aws cloudformation create-stack --stack-name gltest-stack --template-body file://CF1.json

1. **Observe the error message**:
   * As expected, you will encounter the following error:

plaintext

Copy code

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

This error occurs because **NoSecGroup** is referenced in the CF1.json file, but there is no such security group defined.

**6. Fix the Issue in CF1.json**

1. **Open CF1.json and navigate to line 183**:
   * You will see that the security group for the Load Balancer is set to **NoSecGroup**, which doesn't exist in the template.
2. **Fix the security group reference**:
   * Go to line 32 and find the security group name: **WebServerSecurityGroup**.
   * Change line 183 to refer to **WebServerSecurityGroup** instead of **NoSecGroup**.
3. **Save and close the CF1.json file**.

**7. Create the CloudFormation Stack Using CF2.json**

1. **Run the following command** to create the stack using CF2.json:

bash

Copy code

aws cloudformation create-stack --stack-name gltest-stack --template-body file://CF2.json

* + **Check the Logs**: You should now see the stack creation proceeding without errors.

**8. Monitor the Stack Creation in AWS Console**

1. **Navigate to the CloudFormation Console**:
   * Go to the **CloudFormation** service in AWS Console.
2. **Select the Stack**:
   * Click on the **gltest-stack** stack that is in progress.
3. **Observe the Stack Creation**:
   * Wait for the stack to be created successfully. You can monitor the progress by checking the **Events** tab.
   * It may take a few minutes for CloudFormation to complete the stack creation.

**9. Verify the Load Balancer URL**

1. **Once Stack Creation is Complete**:
   * After the stack is created, navigate to the **Outputs** tab in the stack details page.
2. **Click on the URL**:
   * There will be a **URL** output (this should point to the Load Balancer you created).
   * Click the URL to verify that **Apache** was installed and the load balancer is working.