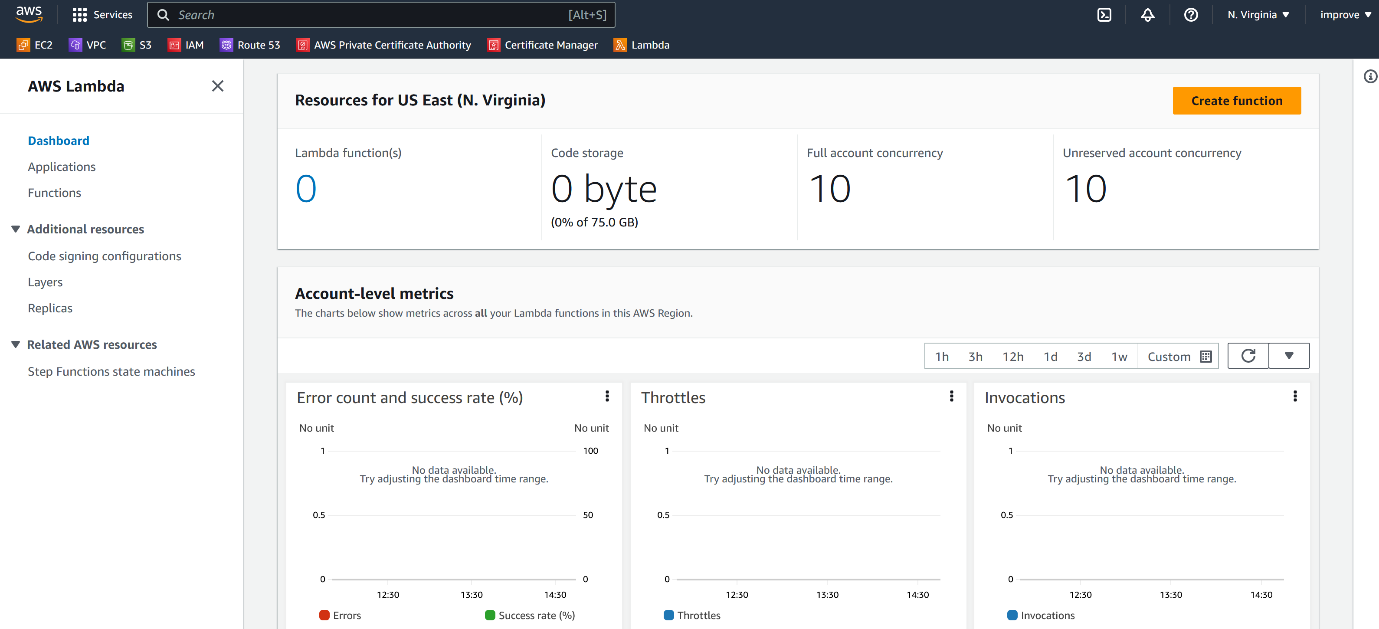
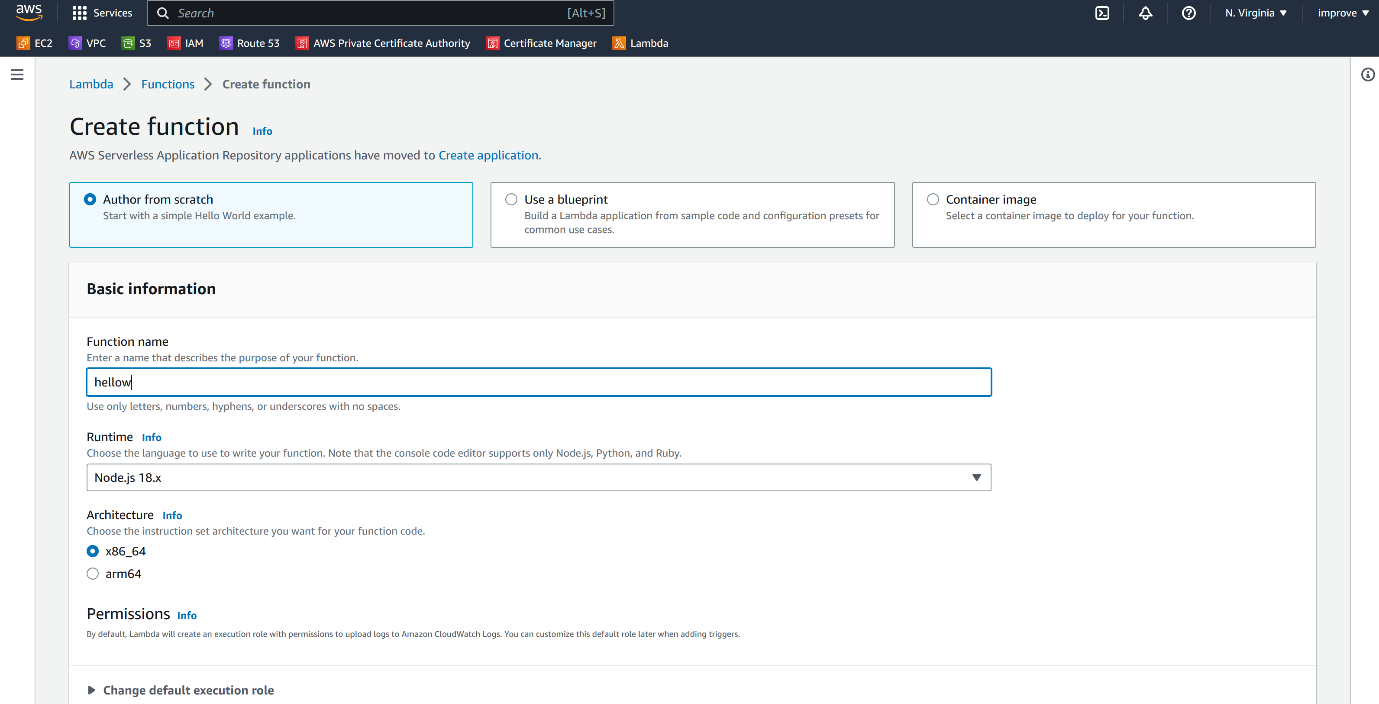
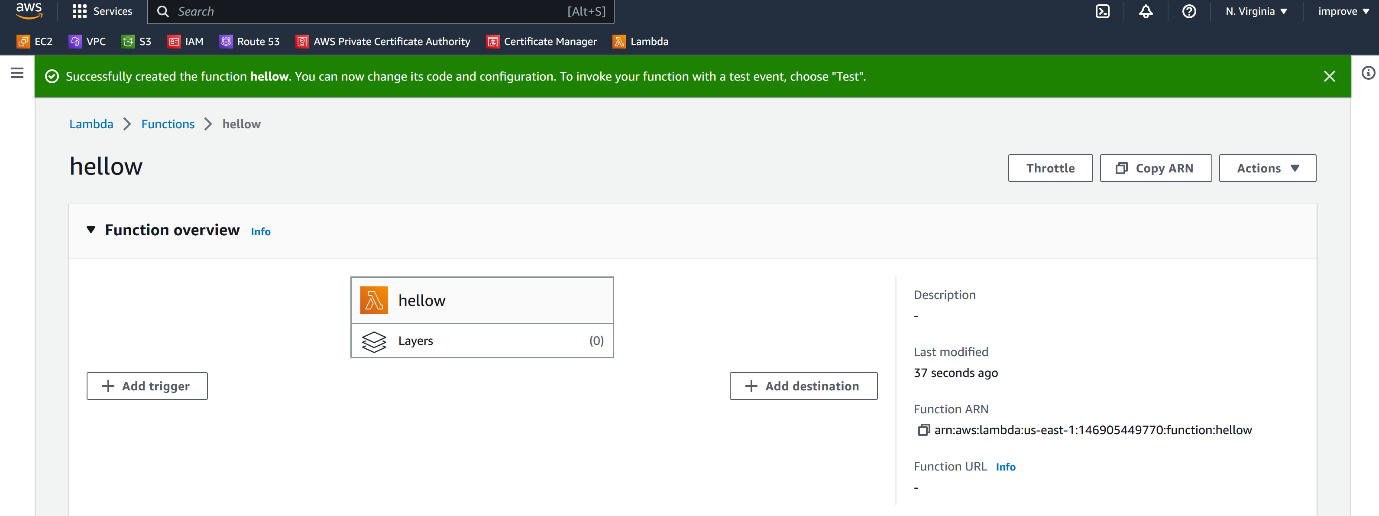
**17. AWS Lambda demo**

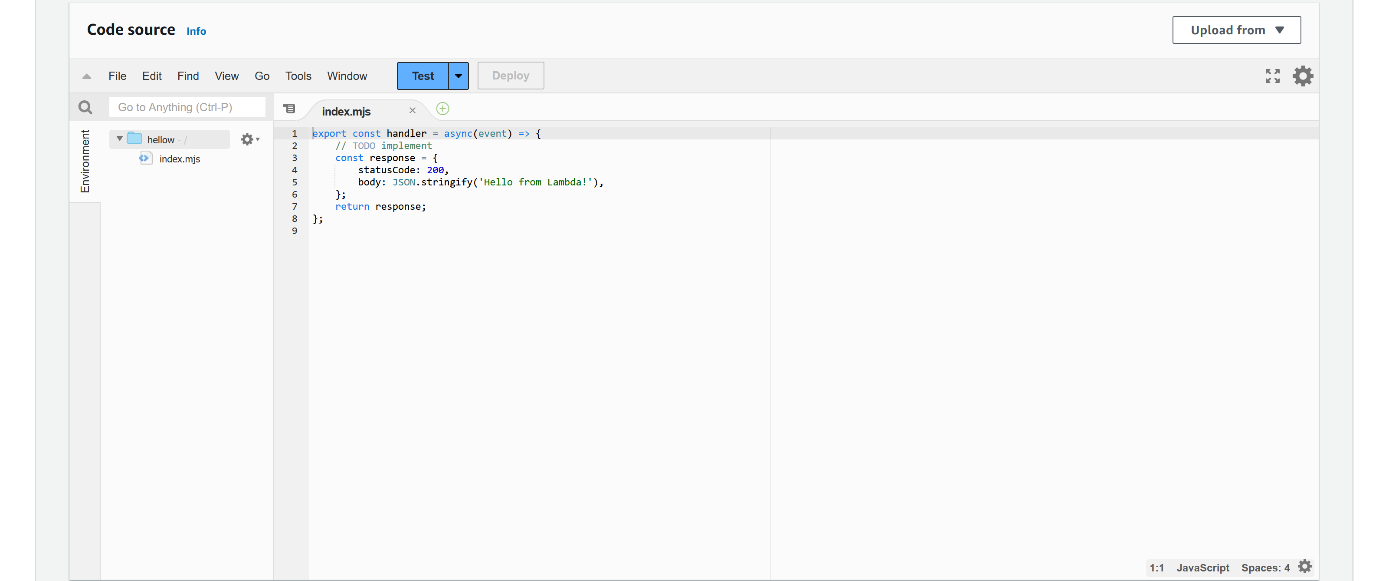


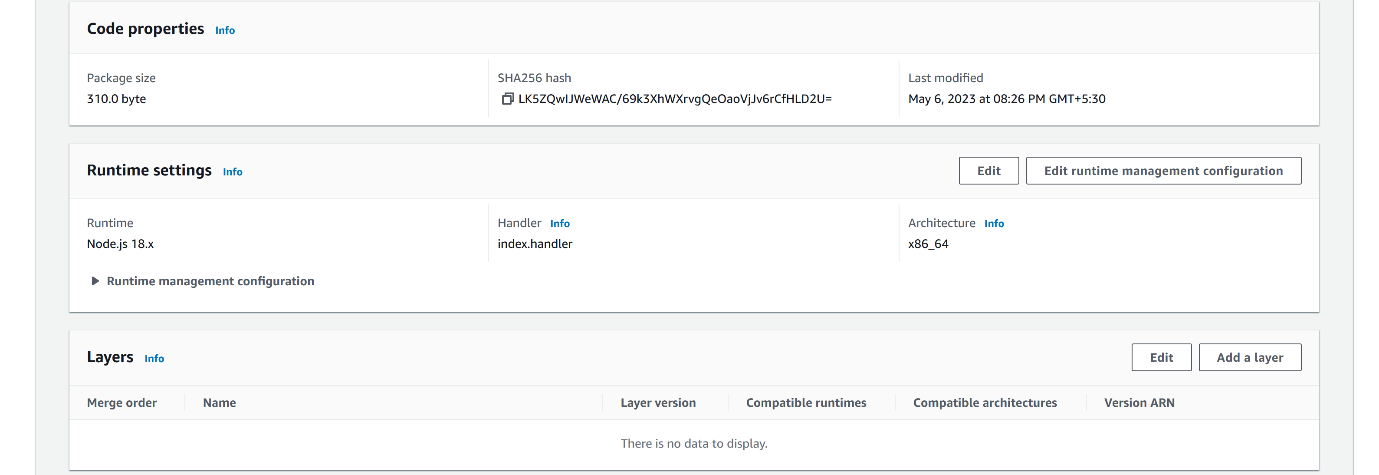
--- click on create.



--- click on create function.

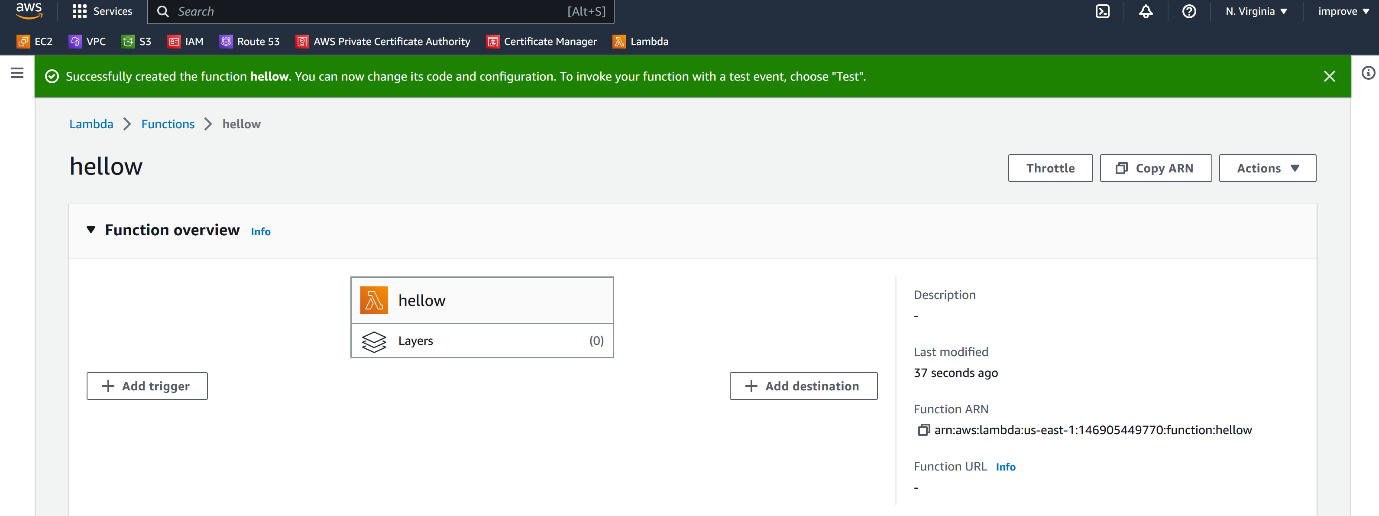




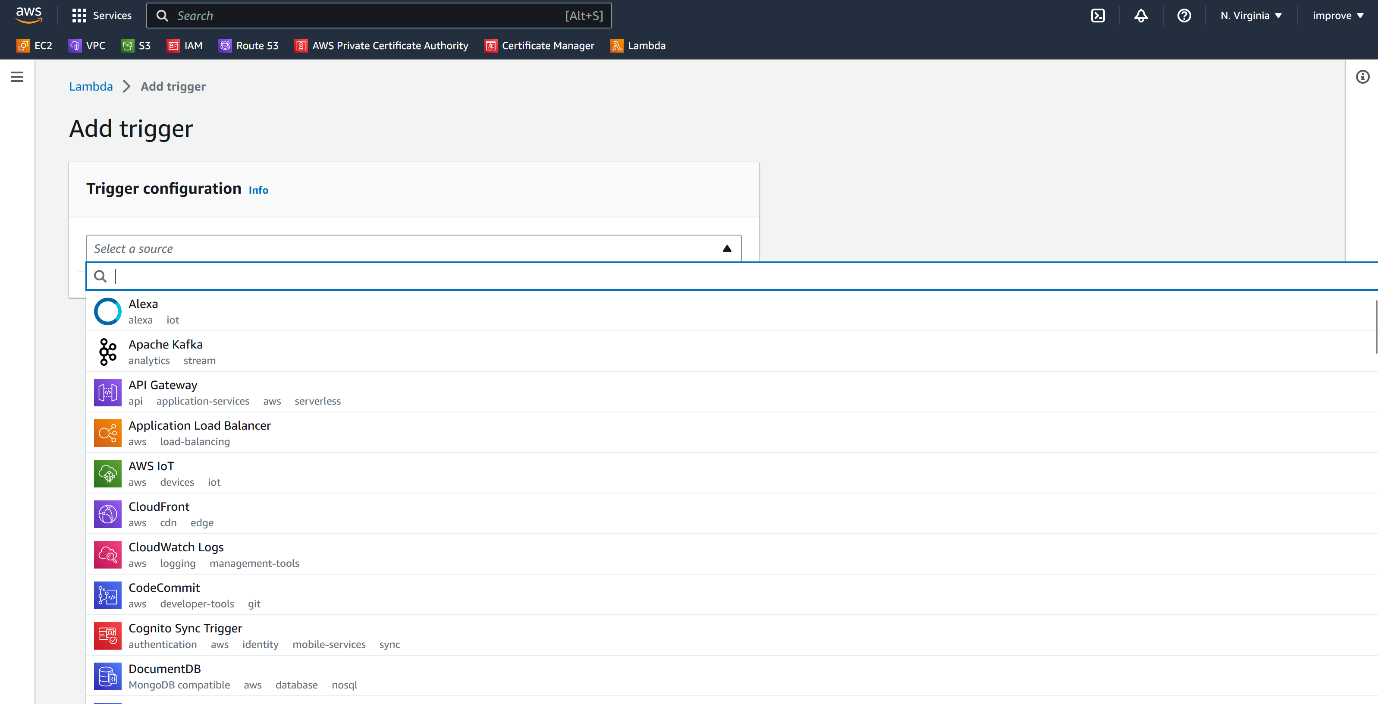


--- aws is added some lambda code in index.mjs.

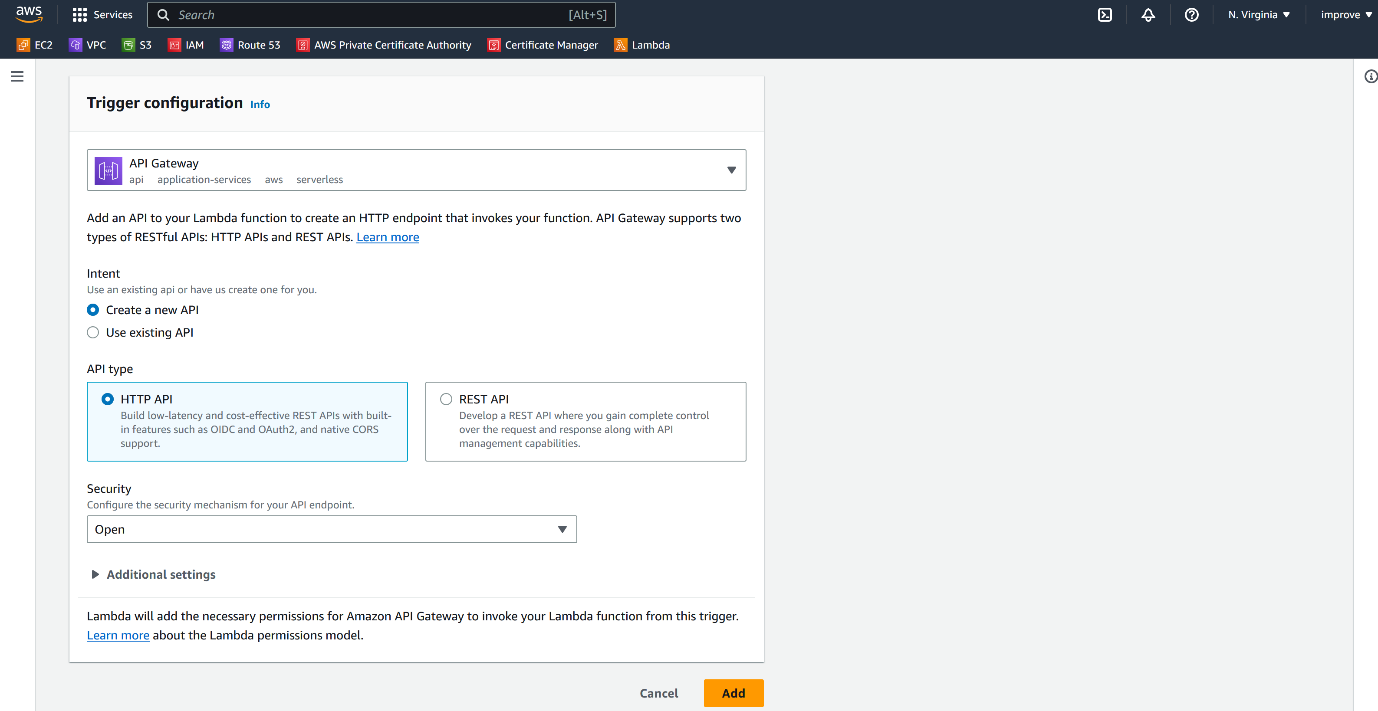
**Add triggers**



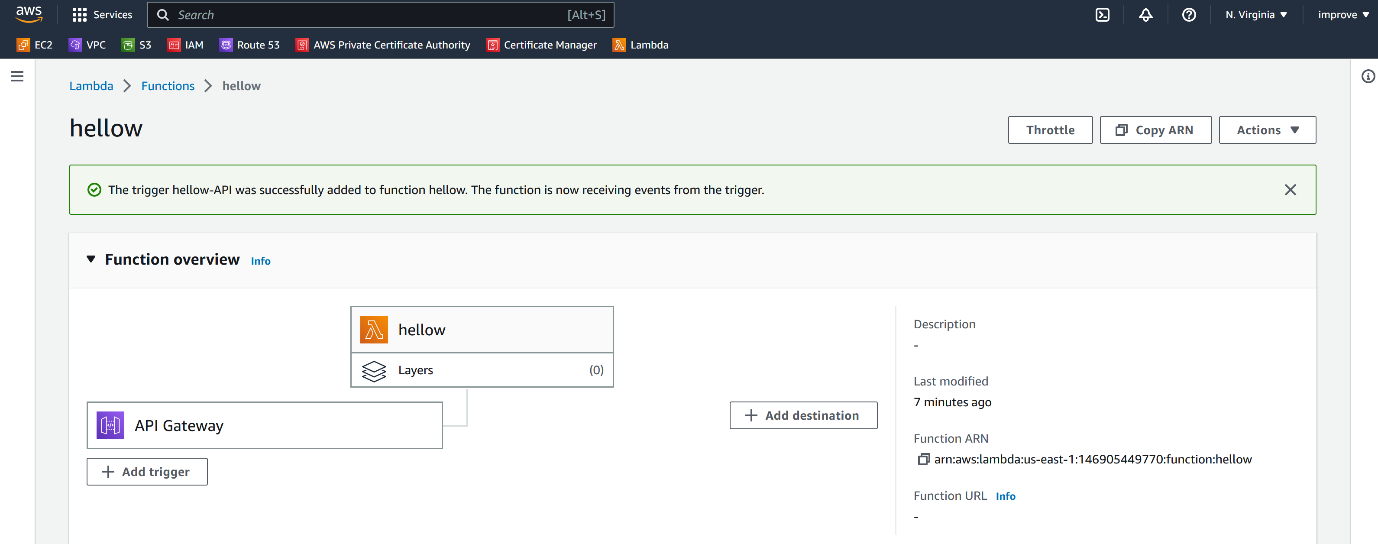
--- click on add triggers.



--- you can add triggers from so many aws lambda services.

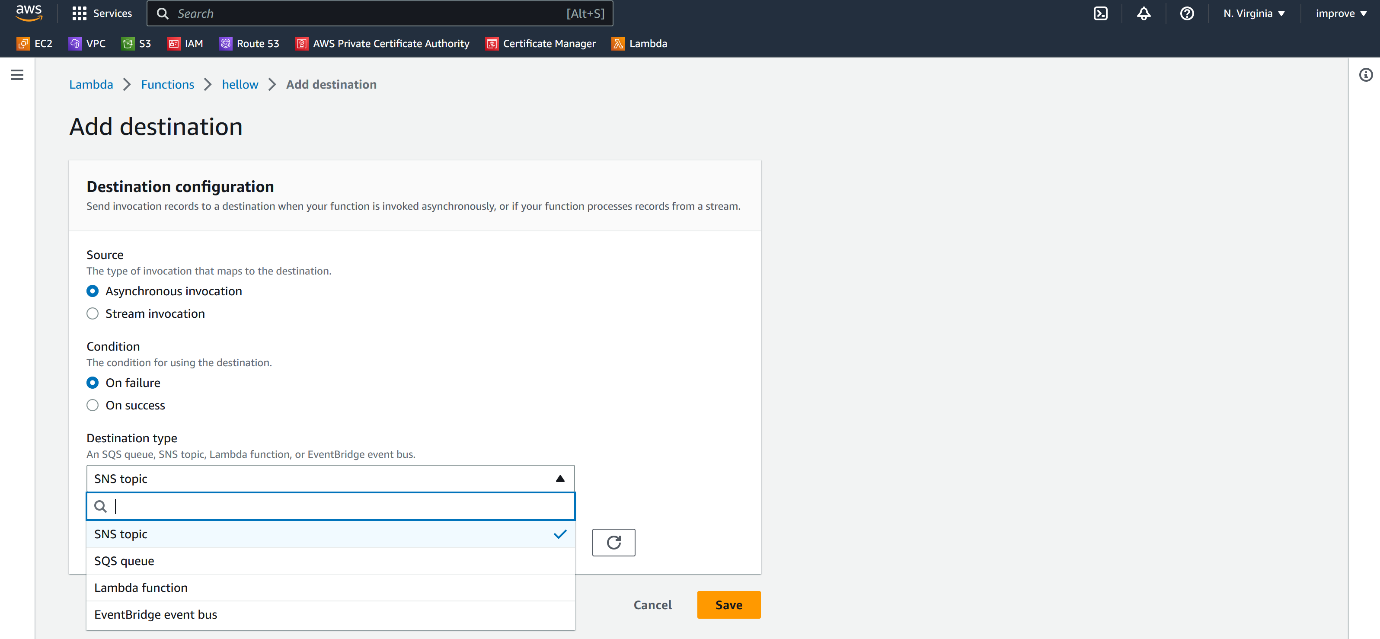


--- here I have selected API Gateway for invocation of lambda function.



--- now you can see, I have setup api gateway to trigger this lambda function. Apart from the trigger, we can also send the output of the lambda function without successful or failure to a different destination.

--- click on add destination.



* Now these are other similar services which we can send the output upon the failure or a success of a lambda functions.
* Here we have used AWS console to create a lambda function but this is not the best practice because if you are building a production application, creating lambda function from a console is not recommended because if you are to create your lambda functions in another AWS account or another environment, you must do all these steps all over again using the AWS console and you probably make mistakes.
* So that is why we are going to use infrastructure's code to provision our lambda functions. You will learn more about it in the next section when we use serverless framework to create our serverless