Create a DeepLens Inference Lambda function

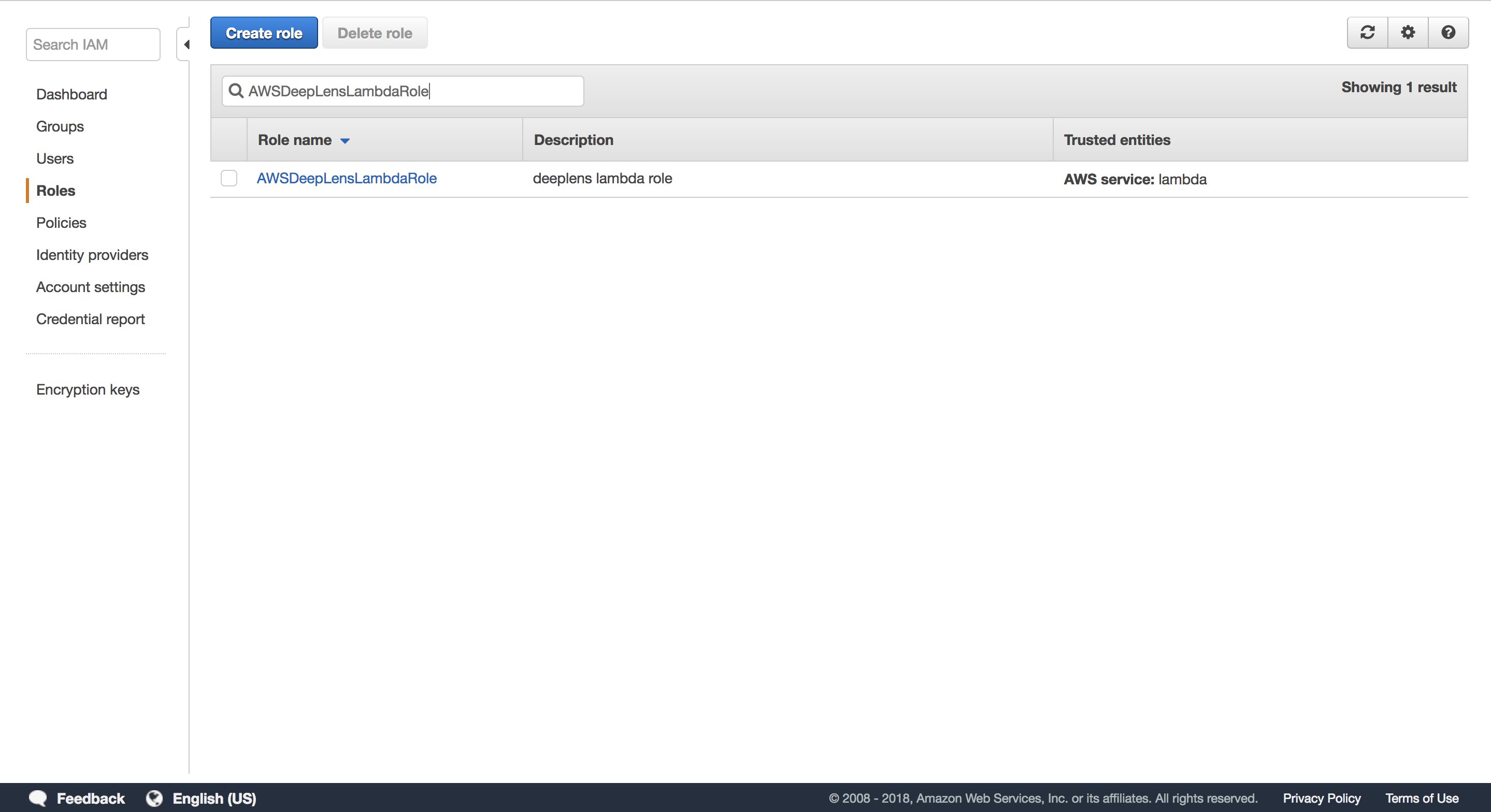
In this module, you will learn to create an inference lambda function for your DeepLens. This lambda function will crop the identified faces and upload them to your S3 bucket.

### IAM

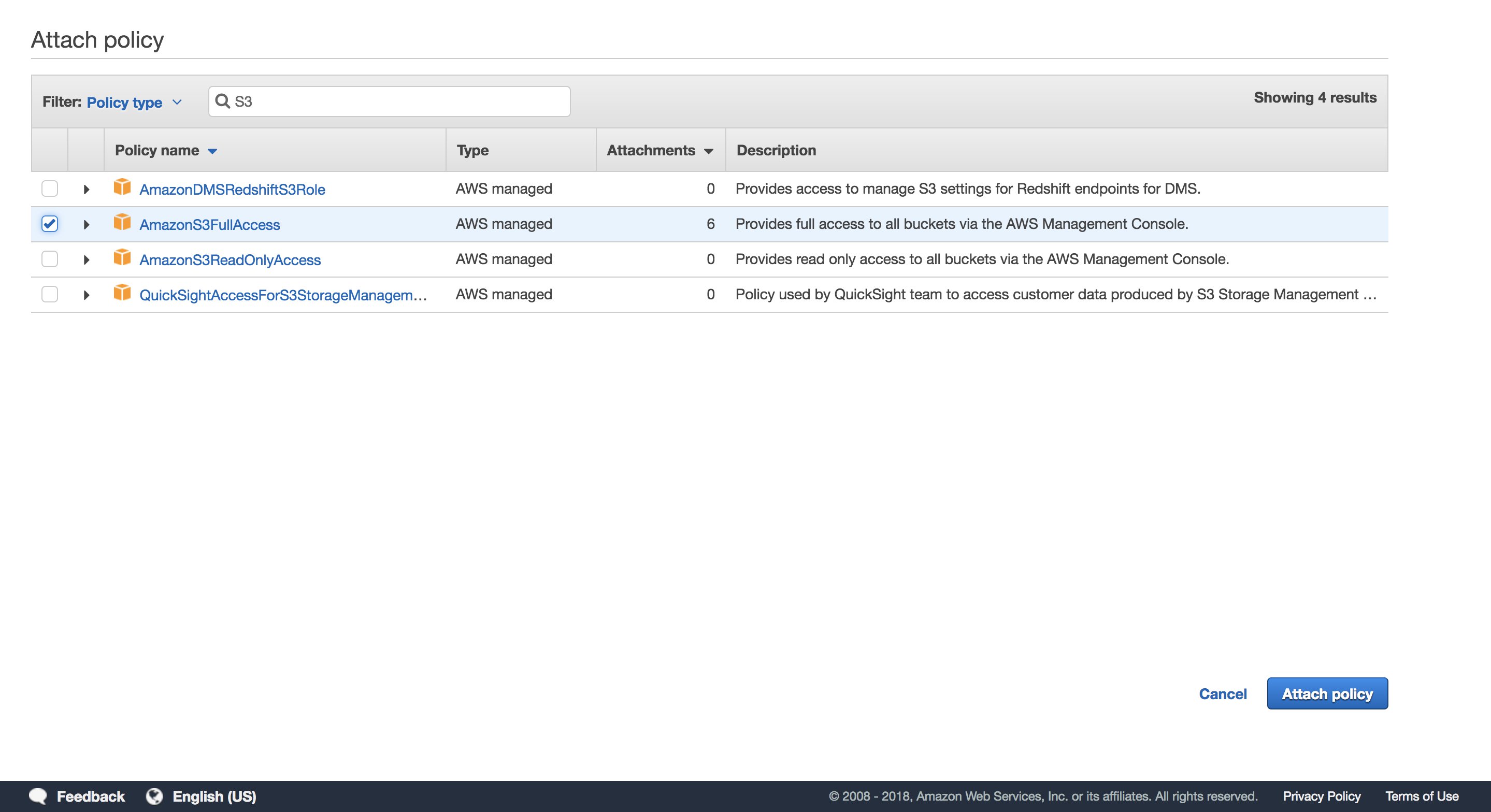
Participants need to first set permissions for the roles the Lambda functions will be using.

First, we need to add S3 permissions to the DeepLens Lambda role so the lambda on the device can call Put Object into the bucket of interest.

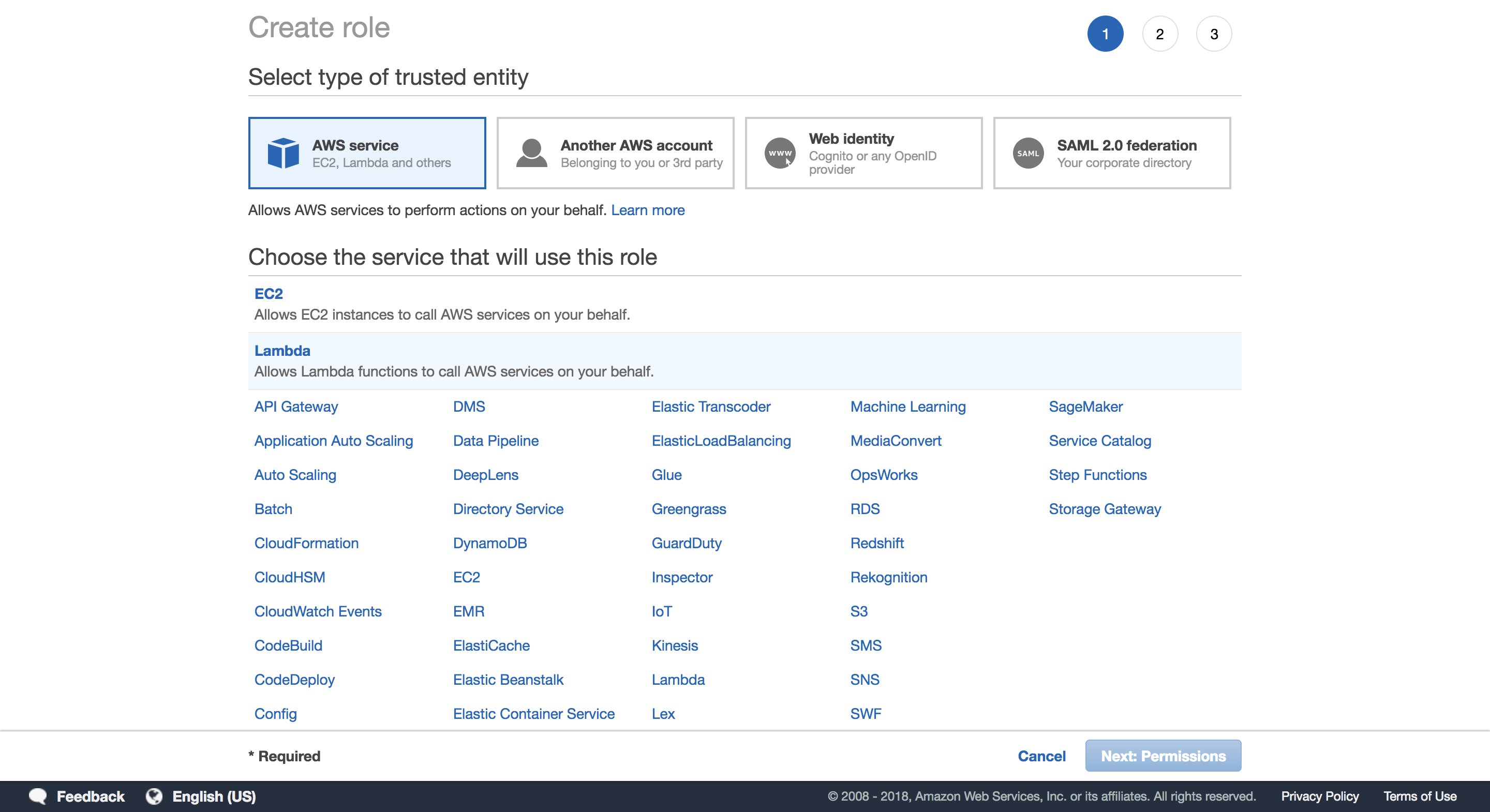
Go to the “Roles” page on the IAM console. Look up the role “AWSDeepLensLambdaRole”



Click on the Role, Click Attach Policy, and attach “AmazonS3FullAccess”



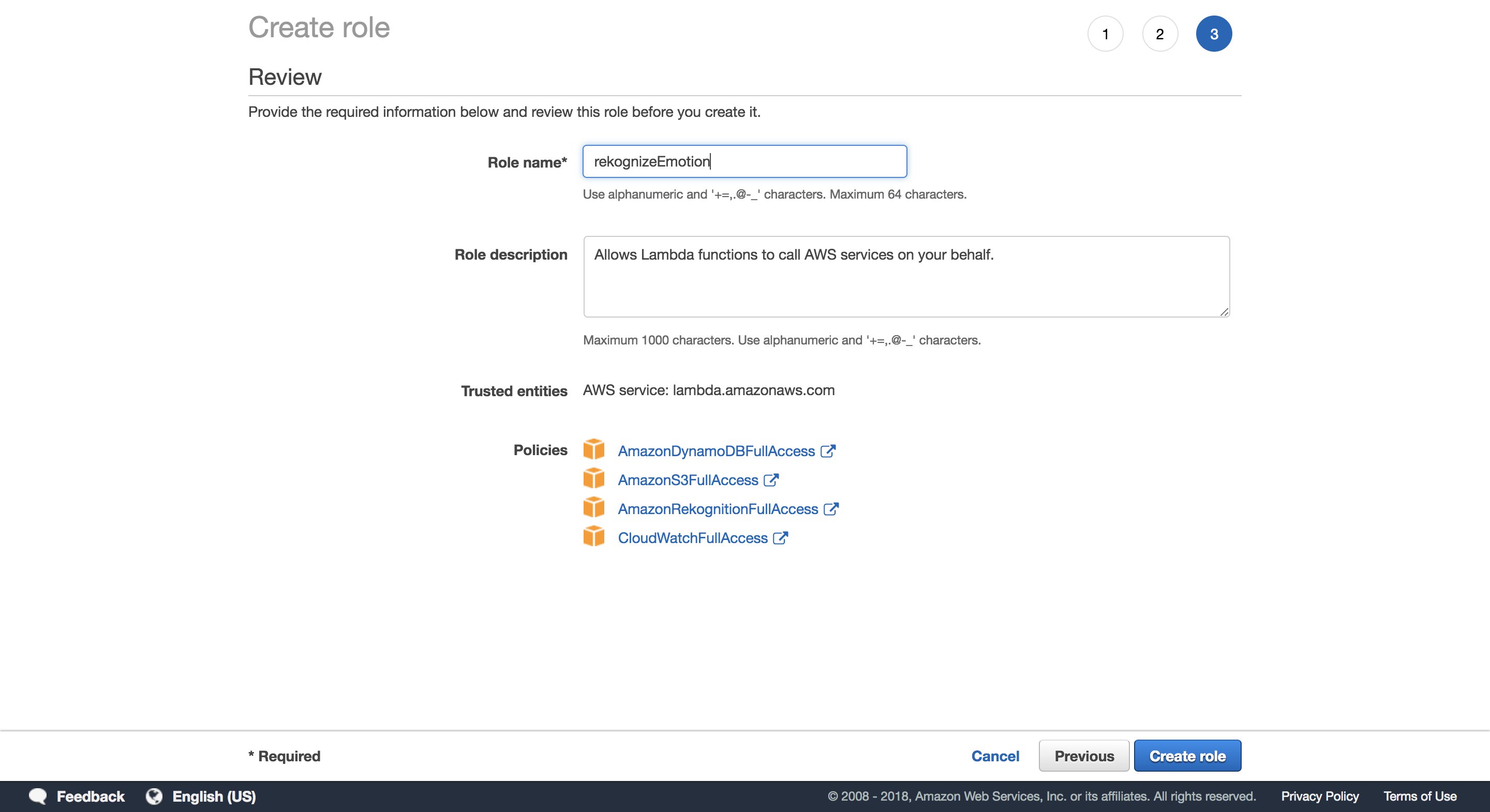
Now we have to create a role for Rekognition Lambda. From “Create Role”, Select “AWS Service” and select “Lambda”:



Attach the following policies:

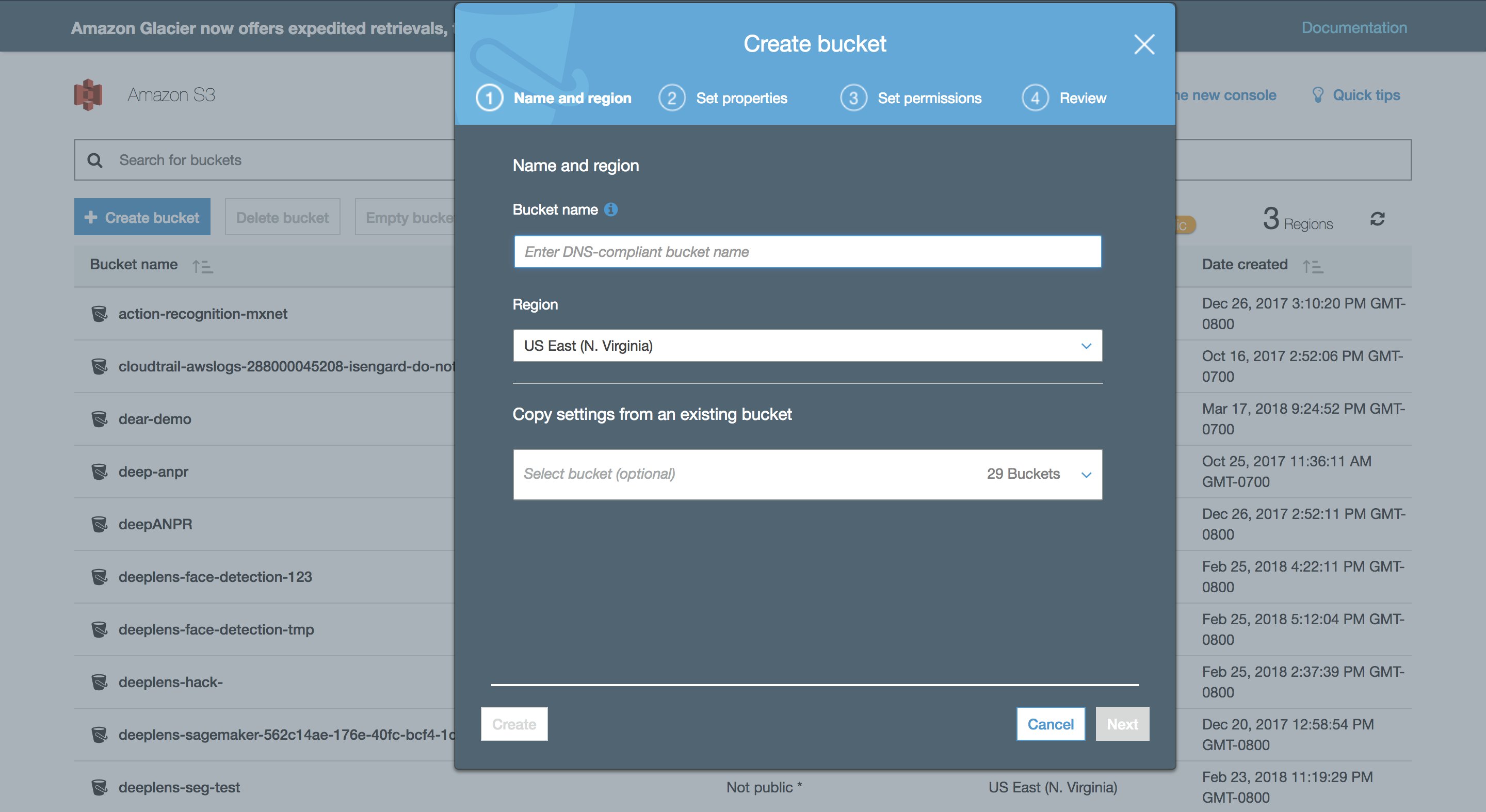
* AmazonDynamoDBFullAcces
* AmazonS3FullAccess
* AmazonRekognitionFullAccess
* CloudWatchFullAccess

And create the role “rekognizeEmotion”:



### S3

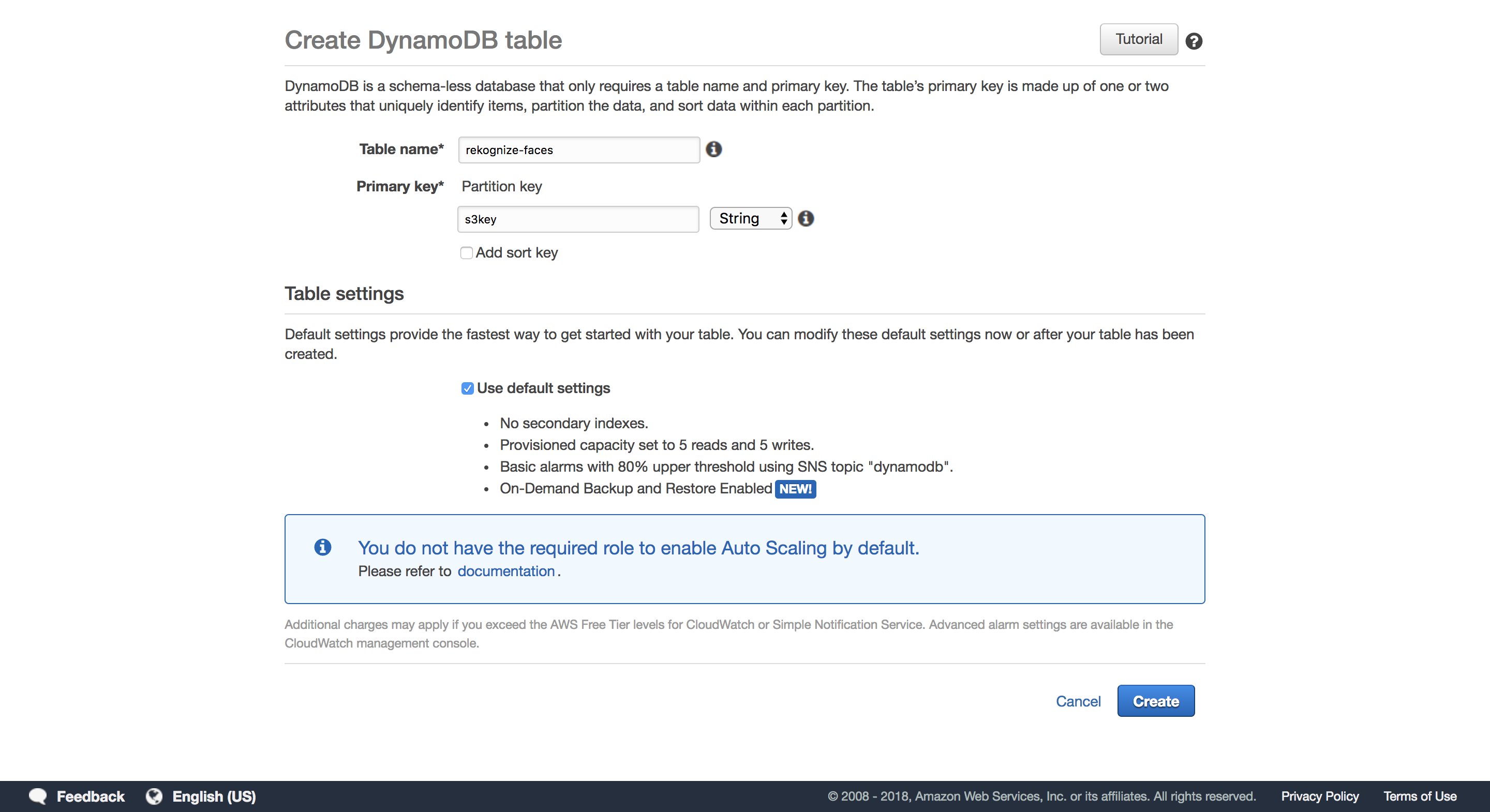
We need to create an S3 bucket that we can upload faces to. These uploads will trigger our Rekognition lambda.



Note the name of the bucket, we'll use it in later steps.

### DYNAModb

Next, we need to create the DynamoDB table that will store our output:



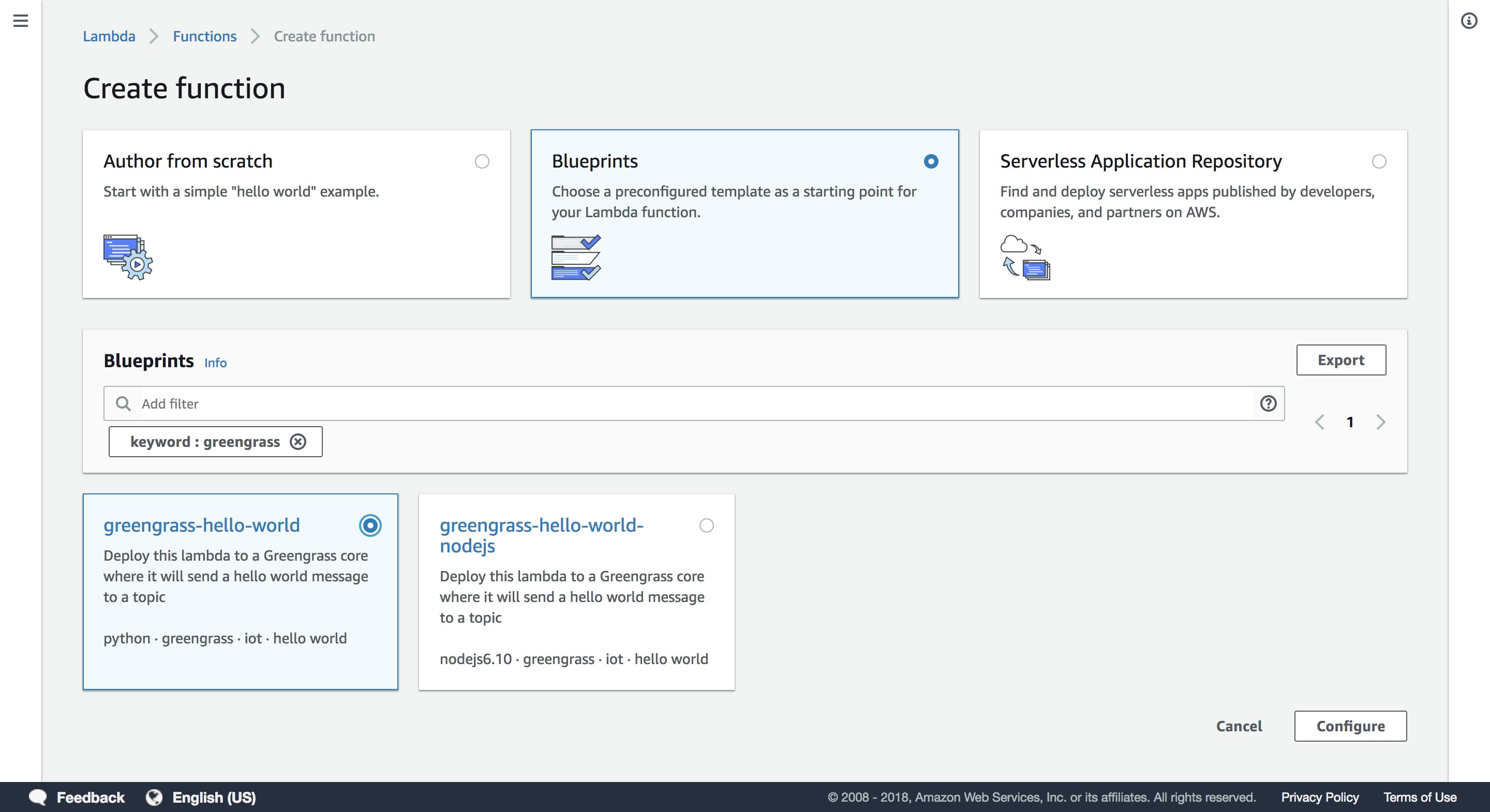
### Lambda Functions

Next, we need to make the two lambda functions:

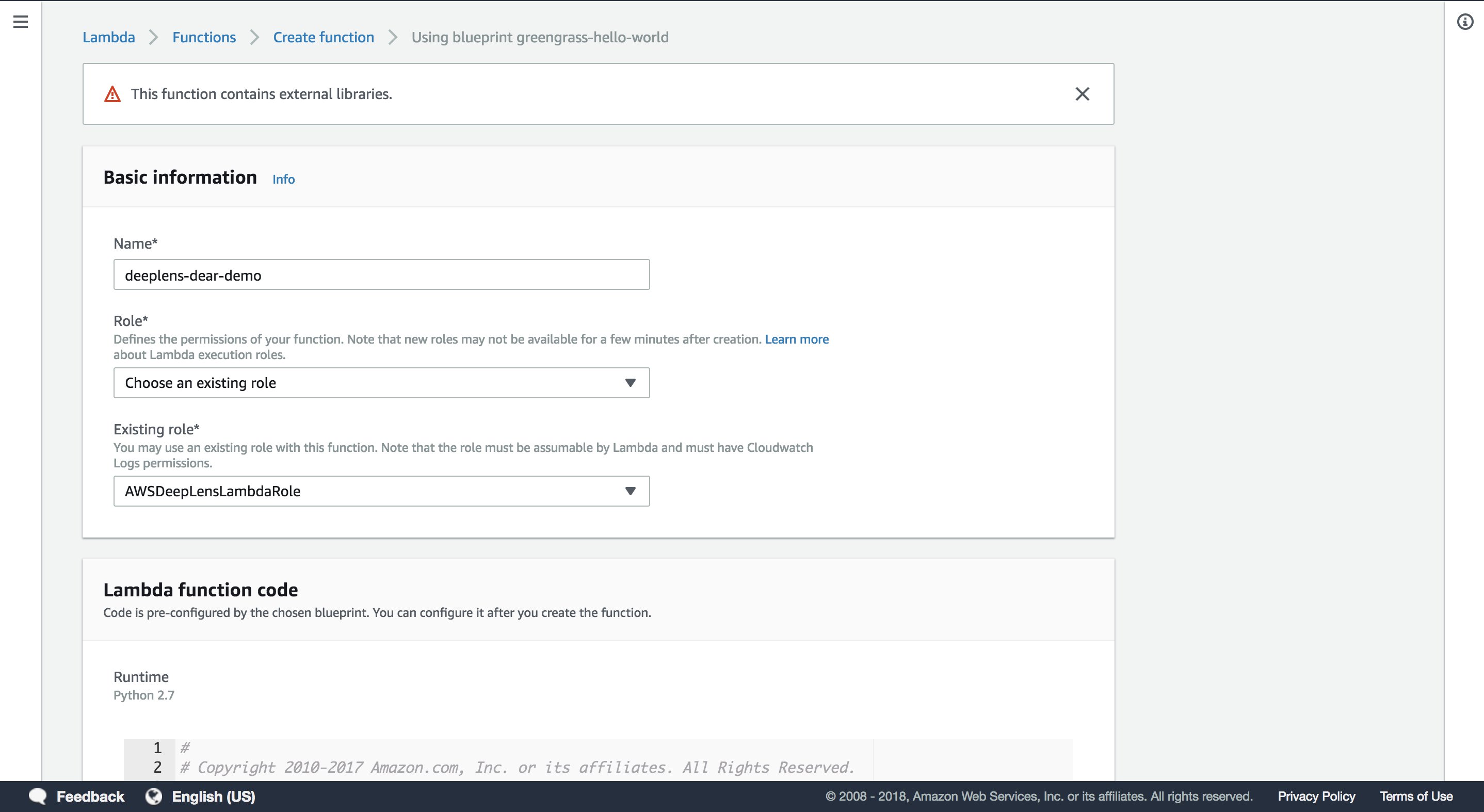
* The greengrass lambda function that runs on the device
* The cloud lambda that sends faces to Rekognition

First, we'll create the greengrass Lambda that will run on the device:

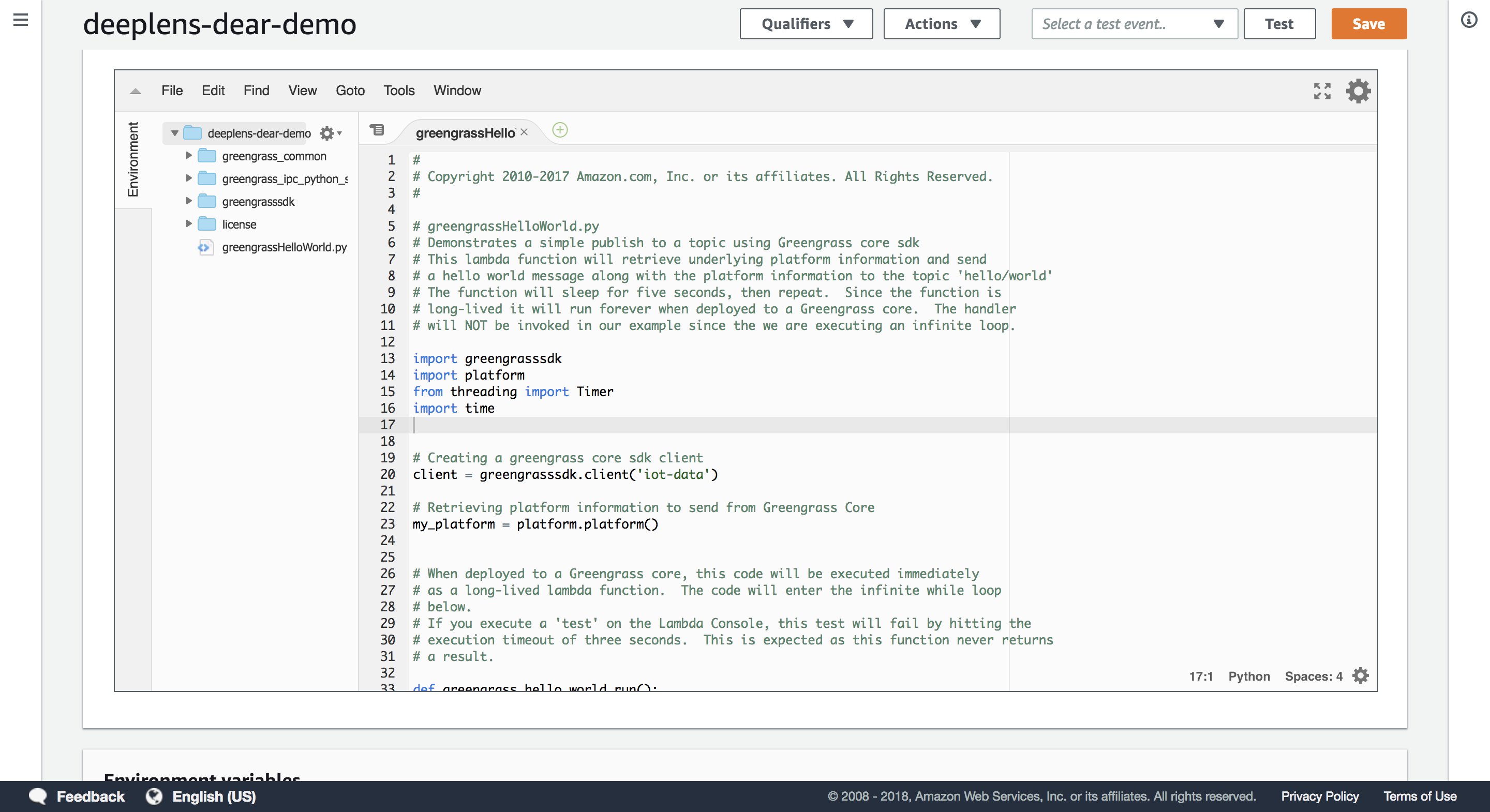
Start by creating a function from the “greengrass-hello-world” blueprint:



Name the function “deeplens-dear-demo”, and attach the AWSDeepLensLambdaRole:

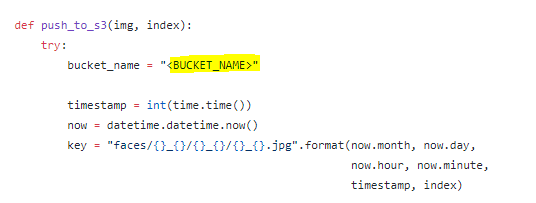


Once created, we're going to replace the default handler code:

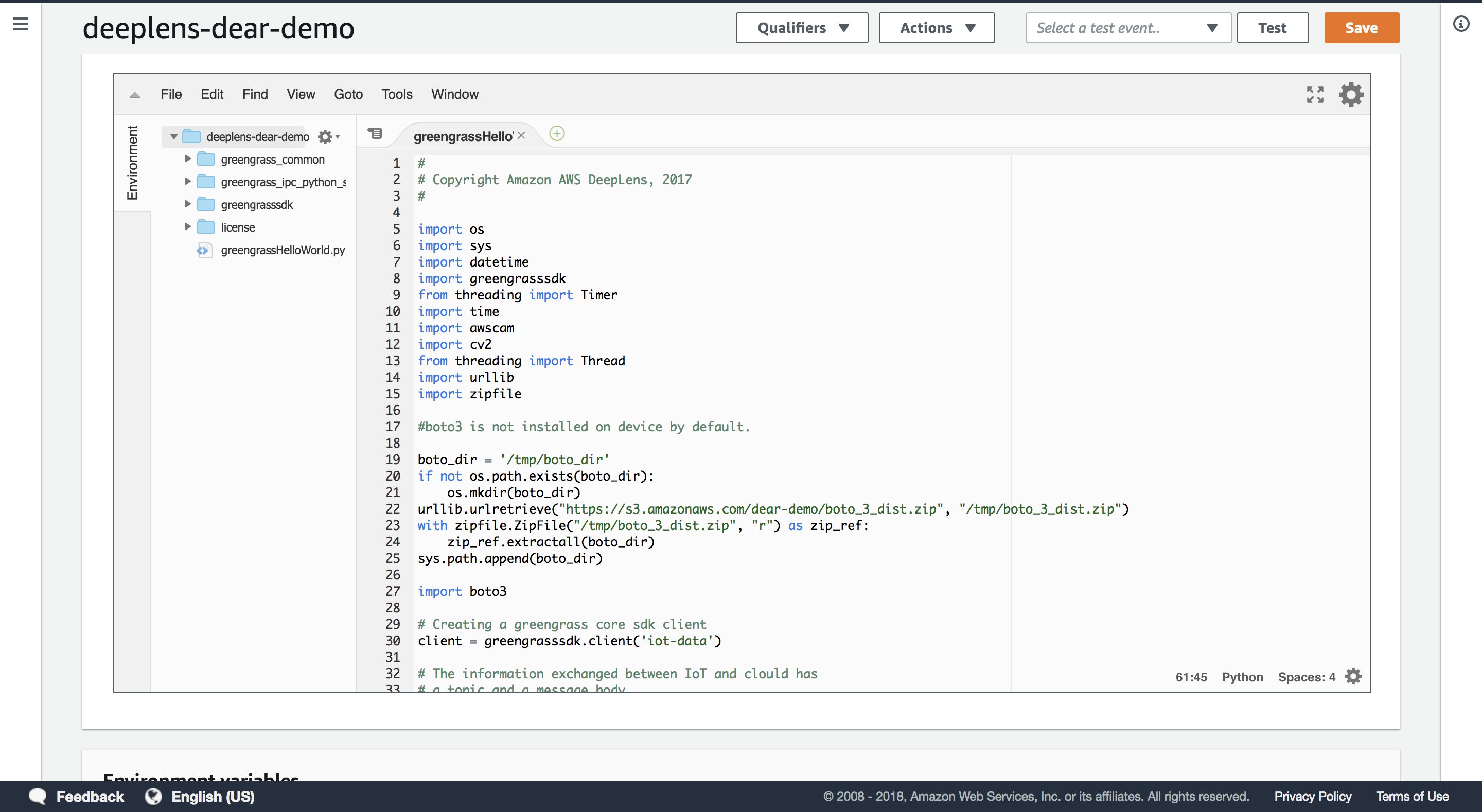


with the inference script (you can find it in the github folder: **inference-lambda.py**), inserting real values for the following fields:

* <BUCKET\_NAME>: the S3 bucket faces will be written to, which we created above



Once replaced:



**Click “Save”, and then under the “Actions” drop-down menu,**

**click “Publish new version” and publish.**