Acessando Recursos AWS no LocalStack

O Lambda precisa de um IAM Role para a sua execução:

```
aws --endpoint http://localhost:4566 \
 --profile localstack \
  iam create-role \
  --role-name lambda-execution \
  --assume-role-policy-document "{\"Version\": \"2012-10-17\",\"Statement\":
[{ \"Effect\": \"Allow\", \"Principal\":
{\"Service\": \"lambda.amazonaws.com\"}, \"Action\": \"sts:AssumeRole\"}]}"
```

Com a IAM Role criada, vamos incluir a policy de execução:

```
aws --endpoint http://localhost:4566 --profile localstack \
  iam attach-role-policy \
  --role-name lambda-execution \
  --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole
```

Efetuamos a implantação do Lambda no LocalStack com o seguinte comando:

```
aws --endpoint http://localhost:4566 --profile localstack \
  lambda create-function \
  --function-name HelloWorld \
  --zip-file fileb://HelloWorld-1.0.jar \
  --handler helloworld.App \
  --runtime java11 \
  --role arn:aws:iam::000000000000:role/lambda-execution
```

Executando o Lambda e verificando se foi instalado corretamente:

```
aws --endpoint http://localhost:4566 --profile localstack \
  lambda invoke \
  --function-name HelloWorld out.txt \
  --log-type Tail
```

Caso tenha executado com sucesso, o arquivo out.txt terá o seguinte conteúdo:

```
{"body":"{ \"message\": \"hello world\", \"location\": \"200.0.0.0\" }","headers":{"Content-
Type":"application/json","X-Custom-Header":"application/json"},"statusCode":200}
```

main.tf # Terraform

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.0"
    }
  }
}
provider "aws" {
  access_key                  = "test"
  secret_key                  = "test"
  region                      = "sa-east-1"
  skip_credentials_validation = true
  skip_metadata_api_check     = true
  skip_requesting_account_id  = true
  endpoints {
    iam     = "http://localhost:4566"
    lambda  = "http://localhost:4566"
  }
}

resource "aws_iam_role" "lambda_execution_role" {
  name = "lambda-execution"
  assume_role_policy = jsonencode({
    Version = "2012-10-17"
    Statement = [
      {
        Effect = "Allow"
        Principal = {
          Service = "lambda.amazonaws.com"
        }
        Action = "sts:AssumeRole"
      }
    ]
  })
}

resource "aws_iam_role_policy_attachment" "lambda_basic_policy_attachment" {
  role       = aws_iam_role.lambda_execution_role.name
  policy_arn = "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole"
```

```
}

resource "aws_lambda_function" "hello_world_function" {
  function_name    = "BookFunction"
  handler          = "bookfunction.controller.BookFunction::handleRequest"
  runtime          = "java11"
  memory_size      = 512
  timeout          = 30
  filename         = "target/HelloWorld-1.0.jar" # Path to the JAR file
  source_code_hash = filebase64sha256("target/HelloWorld-1.0.jar")
  role             = aws_iam_role.lambda_execution_role.arn
}
```

Testando:

```
aws --endpoint http://localhost:4566 --profile localstack lambda invoke --function-name
BookFunction out.txt --log-type Tail


aws --endpoint http://localhost:4566 --profile localstack \
  lambda invoke \
  --function-name BookFunction out.txt \
  --payload '{"httpMethod": "GET", "path": "/"}' \
  --cli-binary-format raw-in-base64-out \
  --log-type Tail
```