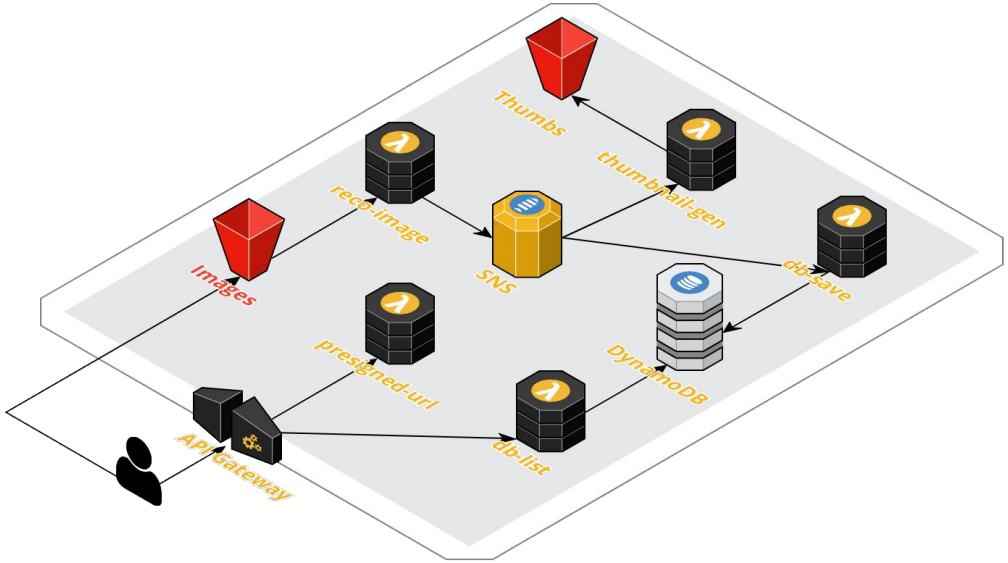
AWStack training - Serverless

Hands On #1 - S3 | IAM - Image bucket

Overview

This Hands-on is composed of 2 parts:

1. S3 Part : new S3 bucket to store images to analyze
2. IAM Part : new IAM role shared by all Lambda function created over the training



Let’s go! | S3 Part

Go to Virginia region N. Virginia

Create a new S3 bucket having these properties:

- **Bucket Name**: serverless-training-img-<xxx>
- **Region**: US East (N.Virginia)

In order to upload images directly from the front-end web-ui, CORS has to be enabled on this bucket:

- Under **Permissions** > **CORS Configuration**, set the xml setting (see Hint 2 for xml template to use)

Once done -> Go to **IAM** Part

Context:

In this part we create the S3 bucket storing the images to be analyzed by the recognition function.

Documentation:

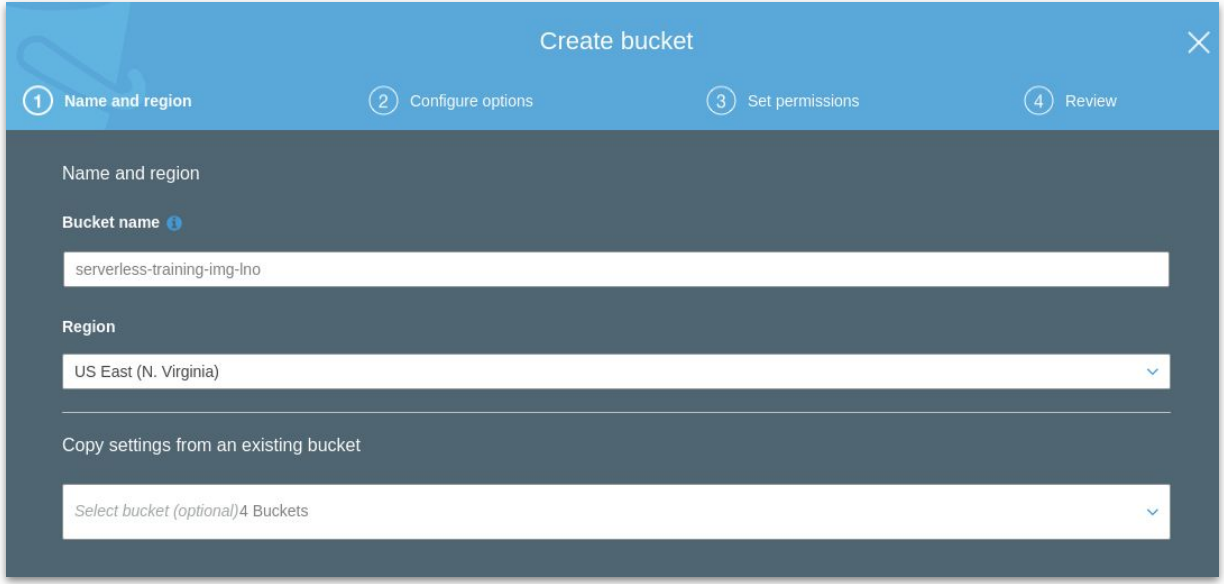
https://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html

Hint 1

S3 - Create a new bucket

“serverless-training-img-<xxx>”

(replace <xxx> by a unique id)



Hint 2

S3 - Enable CORS from “Permissions” > “CORS Configuration”

```xml
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
<CORSRule>
   <AllowedOrigin>*</AllowedOrigin>
   <AllowedMethod>POST</AllowedMethod>
   <MaxAgeSeconds>3000</MaxAgeSeconds>
   <AllowedHeader>*</AllowedHeader>
</CORSRule>
</CORSConfiguration>
```

Let’s go! | IAM Part

Under IAM service, create a new role:

- **Type of trusted entity**: AWS Service
- **Service**: Lambda
- Create a **New Policy** using the **JSON editor** (see Hint 5 for JSON template to use)
- **Policy Name**: serverless_lambda_policy
- **Role Name**: serverless_lambda_role

Context:

In this part we create the IAM role which will be used by all lambda functions created over further Hands-on.

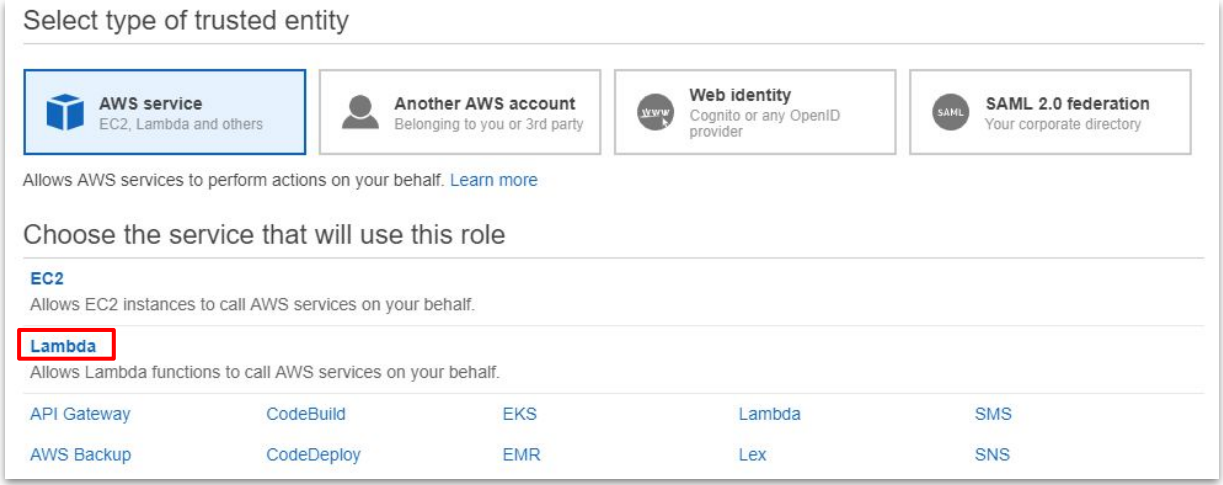
Remark: this is not a best practice to create a unique role gathering all permissions. In actual production applications, we’d rather create a new role for each Lambda function.

Documentation:

https://docs.aws.amazon.com/fr_fr/IAM/latest/UserGuide/introduction.html

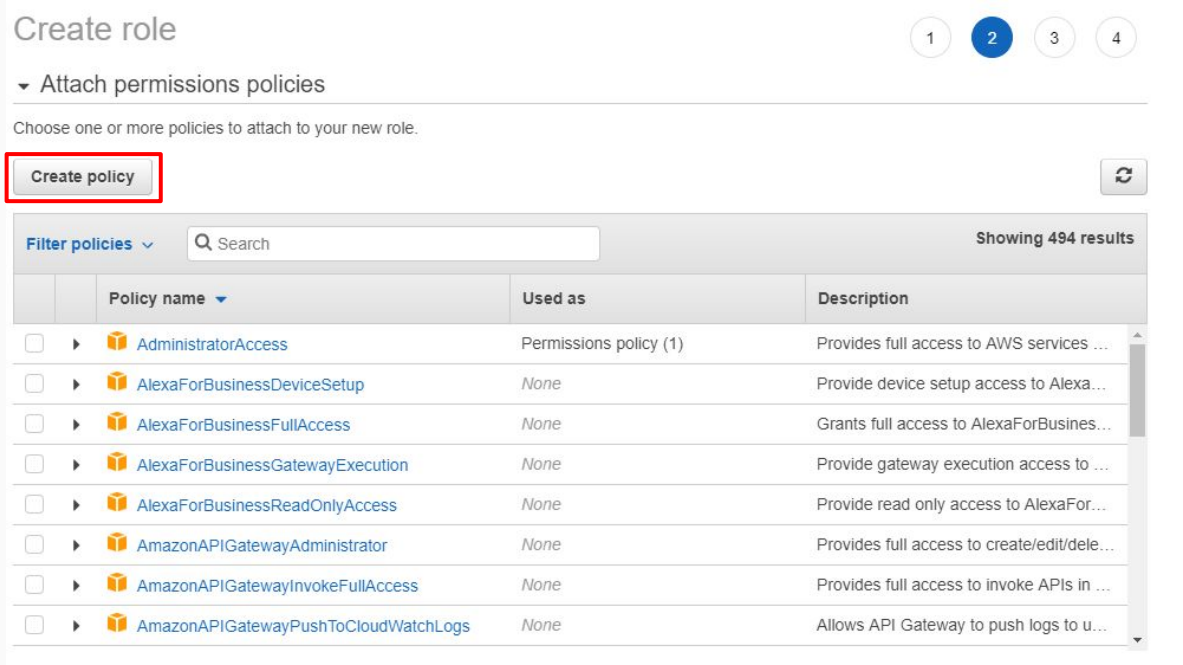
Hint 3

IAM Role Creation - Create a new AWS Service Lambda role



Hint 4

IAM Role Creation - Create a new policy



Hint 5

IAM Policy Creation - Open the "JSON" tab editor and paste the following policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "lambda:GetLayerVersion",
                "lambda:GetLayerVersionPolicy",
                "s3:*",
                "sns:*",
                "dynamodb:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Hint 6

IAM Policy Creation - Review and create policy "serverless_lambda_policy"

Review policy

Name* serverless_lambda_policy

Use alphanumeric and '+=,.@-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

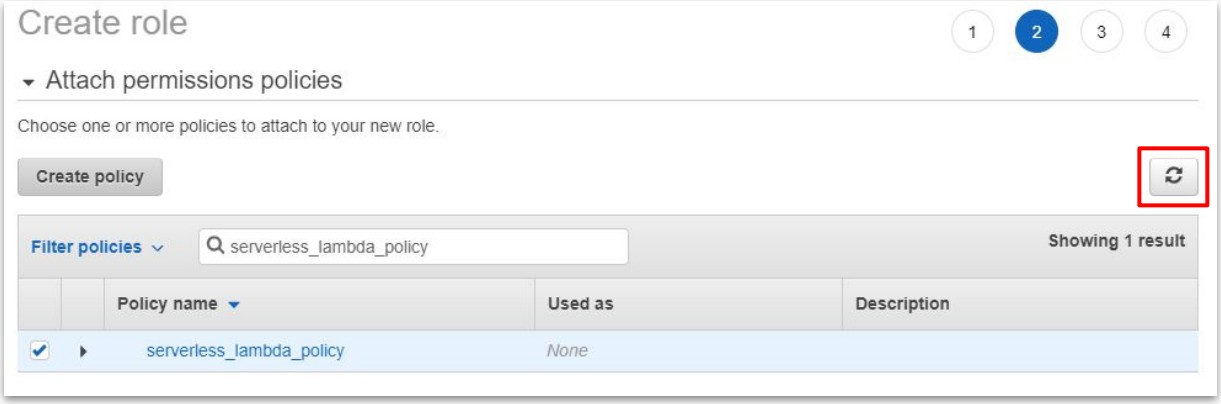
Summary

Filter

Service	Access level	Resource	Request condition
Allow (5 of 170 services) Show remaining 165			
CloudWatch Logs	**Limited**: Write	All resources	None
DynamoDB	Full access	All resources	None
Lambda	**Limited**: Read	All resources	None
S3	Full access	All resources	None
SNS	Full access	All resources	None

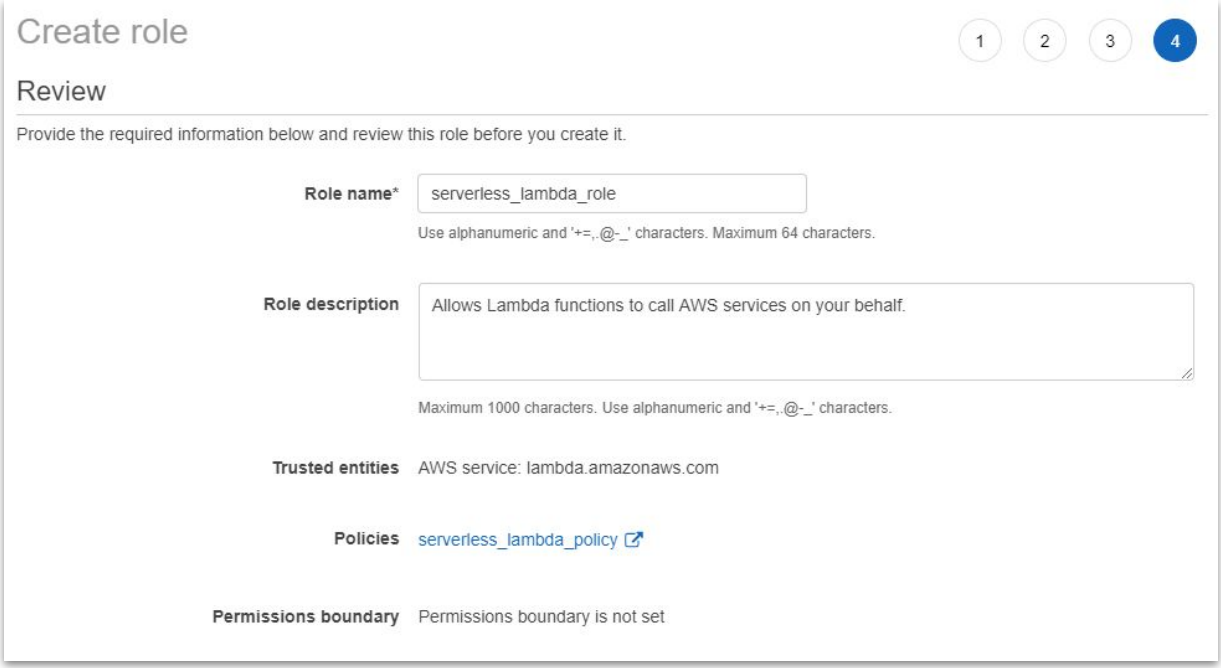
Hint 7

IAM Role Creation - Go back on role creation page, refresh policy list and select the serverless_lambda_policy just created



Hint 8

IAM Role Creation - Assign the role name serverless_lambda_role, review and create it



Done !

No test to perform for this Hands-On !