

# LAB1 - IAM Role for the EC2

The following guide will walk through configuring Identity and Access Management Roles inside of Amazon Web Services for EC2 instance. This Role will grant access for moving data between server and S3 bucket.

## AWS Account

Prior to begin make sure you have created an account with AWS, have entered the relevant billing details and have access to [console.aws.amazon.com](https://console.aws.amazon.com).

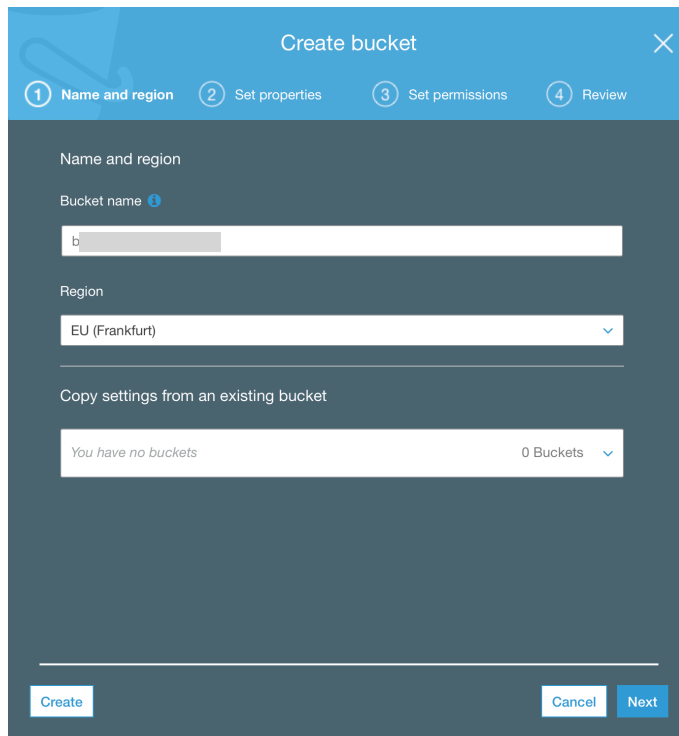
## Create S3 bucket

Create your own S3 bucket which will be accessible from your server.

---

### Step 1. Create S3 bucket.

- Login in to AWS console with your own account
- Go to Services -> Storage -> S3 -> Create bucket (button)
- Enter bucket name, chose a preferred Region and press Create



The screenshot shows the 'Create bucket' wizard in the AWS Management Console. The title bar is blue with a close button (X) on the right. Below the title bar is a progress bar with four steps: 1. Name and region (active), 2. Set properties, 3. Set permissions, and 4. Review. The main content area is dark grey. It has a section 'Name and region' with a 'Bucket name' label and a text input field containing 'b'. Below that is a 'Region' label and a dropdown menu showing 'EU (Frankfurt)'. There is a section 'Copy settings from an existing bucket' with a dropdown menu showing 'You have no buckets' and '0 Buckets'. At the bottom, there are three buttons: 'Create' (white with blue border), 'Cancel' (white with blue border), and 'Next' (blue with white text).

# Setup IAM policy and role for your own account

Before you launch an EC instance, create the IAM role and it's corresponding policy.

---

## Step 2. Create an IAM Policy

- Go to Services -> IAM -> Policies -> Create policy
- Switch to JSON tab and paste the code from S3\_Access\_Policy.json file in Lab1 folder
- In resource section paste the name S3 bucket you just created.
- Press Review policy
- Enter a Name and Description and press Create Policy

---

## Step 3. Create an IAM Role

- Chose a Roles from the menu on the left -> Create role
- Choose the service that will use this role
- Select EC2 and EC2 for your use case
- Press Next: Permissions
- In Search box insert a name of Policy created in previous step
  - (or change Filter to Customer managed if you don't remember the name)
- Select your policy and prest Next: Review
- Enter Role name, Description and press Create role

Role name\*

Maximum 64 characters. Use alphanumeric and '+=, @-\_' characters.

Role description

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

Trusted entities AWS service: ec2.amazonaws.com

Policies  

---

## Step 4. Launch a new EC2 instance

- Go to Services -> EC2
- On the top right corner select your preferred Region
- Press Launch Instance
  - Select Amazon Linux AMI,

- Instance type: t2.micro, Next Configure Instance Details
  - Number of instances: 1
  - Network: default vpc
  - Subnet: leave No preference
  - Auto-assign Public IP: Use subnet setting (Enable)
  - IAM Role: select your role created in Step 3
  - Leave rest options as default and press Add Storage
  - Leave default options and press Next
  - Optionally add tags and press Next
  - Create a new security group with access to over ssh
  - Enter a Name, Description for security group and press Review and Launch
  - Launch an instance and create or chose exist ssh key pair
- Select your instance and press Connect button.
- Depends of your platform (Windows, Linux), connect to the EC2 instance over ssh.

---

## Step 5. Check access to your S3 bucket

Amazon AMIs are provided with pre-installed an aws command line. In this lab, you will use aws cli to work with S3.

- Connect to your EC2 instance over ssh
- Check the access to the s3 by typing: [aws s3 ls](#) and check the result
  - (you should get error: *An error occurred (AccessDenied) when calling the ListBuckets operation: Access Denied*)

```
[ec2-user@ip-172-31-40-141 ~]$ aws s3 ls

An error occurred (AccessDenied) when calling the ListBuckets operation: Access Denied
[ec2-user@ip-172-31-40-141 ~]$
```

- This time typ: [aws s3 ls s3://your\\_bucket\\_name](#)
  - (this time there should be no errors)

---

## Step 6. Create and copy some data to S3 bucket

- In the console of a EC2 create a file: [touch file\\_name](#)
- Copy a file into your s3 bucket: [aws s3 cp file\\_name s3://your\\_bucket/](#)
- List your bucket: [aws s3 ls s3://your\\_bucket](#)

```
[ec2-user@ip-172-31-40-141 ~]$ touch file_doroszl
[ec2-user@ip-172-31-40-141 ~]$ ll
total 0
-rw-rw-r-- 1 ec2-user ec2-user 0 Dec  6 11:59 file_doroszl
[ec2-user@ip-172-31-40-141 ~]$
[ec2-user@ip-172-31-40-141 ~]$ aws s3 cp file_doroszl s3://[redacted]/
upload: ./file_doroszl to s3://[redacted]/file_doroszl
[ec2-user@ip-172-31-40-141 ~]$ aws s3 ls s3://[redacted]
2017-12-06 12:01:29      0 file_doroszl
[ec2-user@ip-172-31-40-141 ~]$ _
```

- Create a folder1 and file1 in that folder:
  - `mkdir folder1`
  - `touch folder1/file1`
- Create a folder1\_1 inside a folder1: `mkdir folder1/folder1_1`
- Copy the folder1 into your S3 bucket: `aws s3 cp --recursive folder1/ s3://your_bucket`
- In the AWS Web Console go to Services -> S3 -> your\_bucket and check the content of the bucket.

The sentence of the commands and each time check the content of your S3 bucket in AWS Web Console

- This time try to copy with a name of the folder: `aws s3 cp --recursive folder1/ s3://your_bucket/folder1/`
- Create a file file1\_1 in folder1\_1: `touch folder1/folder1_1/file1_1`
- Copy to S3: `aws s3 cp --recursive folder1/ s3://your_bucket/folder1/`
- Create another file in folder1\_1: `touch folder1/folder1_1/file1_2`
- This time synchronise folders: `aws s3 sync folder1/ s3://your_bucket/folder1/`

Do you see the difference between cp and sync options?

---

## Step 7. Grant access to view all S3 bucket list.

- Go to Services -> IAM -> Policies and find your policy
- Select your policy, press Edit policy and switch to JSON tab
- In line 19 add “,” after “}”
- Hit enter and paste this additional code:

```
{
  "Effect": "Allow",
  "Action": "s3:ListAllMyBuckets",
  "Resource": "arn:aws:s3:::"
}
```

- (or copy the all policy from S3\_Access\_Policy\_ext.json)
- Press Review and then Save
- Jump back again into you EC2 console and check the command: `aws s3 ls`

Do you still get an error?

---

## Step 8. Create a second bucket.

- From the AWS Web Console create a second bucket
- Jump to your EC2 ssh console and list all buckets: `aws s3 ls`
- Try to list a content of the second bucket: `aws s3 ls s3://your_second_bucket`

What is a result?