

Referência: <https://medium.com/@haydnjmorris/uploading-photos-to-aws-s3-getting-started-with-cognito-and-iam-c96ba5b5496d>

- 1 - No Amazon Cognito criar novo Grupo de Identidade - <https://console.aws.amazon.com/cognito/federated/?region=us-east-1>
- 2 - Editar grupo de entidade:
  - Habilitar acesso para identidades não autenticadas
  - Verifique o nome da função não autenticada, ela é a função do IAM usada na permissão do bucket

**Identities federated**

### Grupo de identidades

Painel  
Código de exemplo  
Navegador de identidade

## Editar grupo de identidades

Nesta página, você pode modificar os detalhes de seu grupo de identidades. Um grupo de identidades deve ter um nome exclusivo e um identidades. Sempre que recebermos uma solicitação para autorizar um usuário, utilizaremos automaticamente as funções especificadas Amazon Cognito. [Saiba mais sobre como usar funções do IAM com o Amazon Cognito.](#)

**Nome do grupo de identidades\***

**ID do grupo de identidades**  (Mostrar ARN)

**Função não autenticada**  [Criar nova função](#)

**Função autenticada**  [Criar nova função](#)

**Identities not authenticated**

O Amazon Cognito pode oferecer suporte a identidades não autenticadas fornecendo um identificador exclusivo e credenciais da AWS para os clientes usem o aplicativo sem fazer login, você poderá habilitar o acesso para identidades não autenticadas. [Saiba mais sobre as ide](#)

Habilitar o acesso para identidades não autenticadas.

### 3 - Permissões do Bucket

```
{
  "Version": "2012-10-17",
  "Id": "Policy1584367912934",
  "Statement": [
    {
      "Sid": "Stmt1584367892876",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::{user-id}:user/{user-name}", //user role ARN
          "arn:aws:iam::{user-id}:role/{unauthRole}" // unauthRole ARN
        ]
      },
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::bucket-name/*"
    }
  ]
}
```

### Android acesso ao Bucket

```
class AwsS3Sync(val activity: Activity): AsyncTask<Any, Any, Any>() {
    override fun doInBackground(vararg params: Any?): Any? { // Initialize the Amazon Cognito credentials provider
        val credentialsProvider =
            CognitoCachingCredentialsProvider(
                activity.applicationContext, // Application Context
                "identity pool id", // Identity Pool ID
            )
    }
}
```

```
Regions.US_EAST_1 // Region enum,
)
val s3Client =
    AmazonS3Client(credentialsProvider, Region.getRegion(Regions.US_EAST_1))
val fileToUpload: File = File(activity.filesDir, "sample.txt")

val writer = FileWriter(fileToUpload)
writer.append("Howdy World!")
writer.close()
// (Replace "MY-BUCKET" with your S3 bucket name, and "MY-OBJECT-KEY" with whatever you would like to name the file in S3)
val putRequest = PutObjectRequest(
    "bucket-name",

    "sample15.txt",
    fileToUpload
)
val putResponse: PutObjectResult = s3Client.putObject(putRequest)

// Log.i("AWS", putResponse.)

val getRequest = GetObjectRequest("bucket-name", "sample15.txt")
val getResponse: S3Object = s3Client.getObject(getRequest)
val myObjectBytes: InputStream = getResponse.getObjectContent()
// Do what you want with the object
myObjectBytes.close()
return null
}
}
```