PUT method : generate presigned url

In the below program, we are generating signed url for uploading a file with name **samplevideo** under **sample** folder. That’s why below objectkey=sample/samplevideo

**package** com.aws.test;

**import** java.io.IOException;

**import** java.io.OutputStreamWriter;

**import** java.net.HttpURLConnection;

**import** java.net.URL;

**import** com.amazonaws.AmazonClientException;

**import** com.amazonaws.AmazonServiceException;

**import** com.amazonaws.HttpMethod;

**import** com.amazonaws.auth.PropertiesCredentials;

**import** com.amazonaws.auth.profile.ProfileCredentialsProvider;

**import** com.amazonaws.services.s3.AmazonS3;

**import** com.amazonaws.services.s3.AmazonS3Client;

**import** com.amazonaws.services.s3.Headers;

**import** com.amazonaws.services.s3.model.CannedAccessControlList;

**import** com.amazonaws.services.s3.model.GeneratePresignedUrlRequest;

**import** com.infotree.qliktest.FileDownloadFromAmazonS3;

**public** **class** GeneratePresignedUrlAndUploadObject {

**private** **static** String *bucketName* = "chalapathisvssr";

**private** **static** String *objectKey* = "sample/samplevideo";

**public** **static** **void** main(String[] args) **throws** IOException {

AmazonS3 s3client = **new** AmazonS3Client(**new** PropertiesCredentials(

FileDownloadFromAmazonS3.**class**.getClassLoader()

.getResourceAsStream("aws.properties")));

**try** {

System.***out***.println("Generating pre-signed URL.");

java.util.Date expiration = **new** java.util.Date();

**long** milliSeconds = expiration.getTime();

milliSeconds += 1000 \* 60 \* 60; // Add 1 hour.

expiration.setTime(milliSeconds);

GeneratePresignedUrlRequest generatePresignedUrlRequest =

**new** GeneratePresignedUrlRequest(*bucketName*, *objectKey*);

generatePresignedUrlRequest.setMethod(HttpMethod.***PUT***);

generatePresignedUrlRequest.setExpiration(expiration);

//generatePresignedUrlRequest.addRequestParameter(Headers.S3\_CANNED\_ACL, CannedAccessControlList.PublicRead.toString());

URL url = s3client.generatePresignedUrl(generatePresignedUrlRequest);

//UploadObject(url);

System.***out***.println("Pre-Signed URL = " + url.toString());

} **catch** (AmazonServiceException exception) {

System.***out***.println("Caught an AmazonServiceException, " +

"which means your request made it " +

"to Amazon S3, but was rejected with an error response " +

"for some reason.");

System.***out***.println("Error Message: " + exception.getMessage());

System.***out***.println("HTTP Code: " + exception.getStatusCode());

System.***out***.println("AWS Error Code:" + exception.getErrorCode());

System.***out***.println("Error Type: " + exception.getErrorType());

System.***out***.println("Request ID: " + exception.getRequestId());

} **catch** (AmazonClientException ace) {

System.***out***.println("Caught an AmazonClientException, " +

"which means the client encountered " +

"an internal error while trying to communicate" +

" with S3, " +

"such as not being able to access the network.");

System.***out***.println("Error Message: " + ace.getMessage());

}

}

**public** **static** **void** UploadObject(URL url) **throws** IOException

{

HttpURLConnection connection=(HttpURLConnection) url.openConnection();

connection.setDoOutput(**true**);

connection.setRequestMethod("PUT");

OutputStreamWriter out = **new** OutputStreamWriter(

connection.getOutputStream());

out.write("This text uploaded as object.");

out.close();

**int** responseCode = connection.getResponseCode();

System.***out***.println("Service returned response code " + responseCode);

}

}

OUT PUT: signed url will be generated.

Take that above generated signed url and upload any file with below code. In the below code I have uploaded video file, which will be saved under sample folder.

// presigned url upload program

File fileToUpload = **new** File("E:\\SampleVideo.mp4");

HttpPut putreq = **new** HttpPut(URI.*create*("https://chalapathisvssr.s3.amazonaws.com/sample/samplevideo?AWSAccessKeyId=AKIAI2EBDM4A7M7G6RXQ&Expires=1512740338&Signature=Y61f%2FA2Do801TYZRVqGMUDKOQCg%3D"));

// AES256 is currently the only supported algorithm for SSE-S3

/\* putreq.addHeader(new BasicHeader(Headers.SERVER\_SIDE\_ENCRYPTION,

SSEAlgorithm.AES256.getAlgorithm()));\*/

putreq.setEntity(**new** FileEntity(fileToUpload));

CloseableHttpClient httpclient = HttpClients.*createDefault*();

httpclient.execute(putreq);

GET Method: Generate presigned url

In the below program, we are generating signed url for accessing the file from given objectkey.

Below program objectkey=sample/samplevideo , we are accessing **samplevideo** file under **sample** folder.

**package** com.aws.test;

**import** java.io.IOException;

**import** java.net.URL;

**import** com.amazonaws.AmazonClientException;

**import** com.amazonaws.AmazonServiceException;

**import** com.amazonaws.HttpMethod;

**import** com.amazonaws.auth.PropertiesCredentials;

**import** com.amazonaws.auth.profile.ProfileCredentialsProvider;

**import** com.amazonaws.services.s3.AmazonS3;

**import** com.amazonaws.services.s3.AmazonS3Client;

**import** com.amazonaws.services.s3.model.GeneratePresignedUrlRequest;

**import** com.infotree.qliktest.FileDownloadFromAmazonS3;

**public** **class** GeneratePreSignedUrl {

**private** **static** String *bucketName* = "chalapathisvssr";

**private** **static** String *objectKey* = "sample/samplevideo";

**public** **static** **void** main(String[] args) **throws** IOException {

AmazonS3 s3client = **new** AmazonS3Client(**new** PropertiesCredentials(

FileDownloadFromAmazonS3.**class**.getClassLoader()

.getResourceAsStream("aws.properties")));

**try** {

System.***out***.println("Generating pre-signed URL.");

java.util.Date expiration = **new** java.util.Date();

**long** milliSeconds = expiration.getTime();

milliSeconds += 1000 \* 60 \* 60; // Add 1 hour.

expiration.setTime(milliSeconds);

GeneratePresignedUrlRequest generatePresignedUrlRequest =

**new** GeneratePresignedUrlRequest(*bucketName*, *objectKey*);

generatePresignedUrlRequest.setMethod(HttpMethod.***GET***);

generatePresignedUrlRequest.setExpiration(expiration);

URL url = s3client.generatePresignedUrl(generatePresignedUrlRequest);

System.***out***.println("Pre-Signed URL = " + url.toString());

} **catch** (AmazonServiceException exception) {

System.***out***.println("Caught an AmazonServiceException, " +

"which means your request made it " +

"to Amazon S3, but was rejected with an error response " +

"for some reason.");

System.***out***.println("Error Message: " + exception.getMessage());

System.***out***.println("HTTP Code: " + exception.getStatusCode());

System.***out***.println("AWS Error Code:" + exception.getErrorCode());

System.***out***.println("Error Type: " + exception.getErrorType());

System.***out***.println("Request ID: " + exception.getRequestId());

} **catch** (AmazonClientException ace) {

System.***out***.println("Caught an AmazonClientException, " +

"which means the client encountered " +

"an internal error while trying to communicate" +

" with S3, " +

"such as not being able to access the network.");

System.***out***.println("Error Message: " + ace.getMessage());

}

}

}

OUTPUT: generates presigned url

If above program signed url is video/image file, then we can access this video file in the browser through below html video tag. then save as html file and open in the browser. Then video will be play.

<!DOCTYPE html>

<html>

<body>

<video width="320" height="240" controls>

<source src="https://chalapathisvssr.s3.amazonaws.com/sample/samplevideo?AWSAccessKeyId=AKIAI2EBDM4A7M7G6RXQ&Expires=1512741563&Signature=pcA7u2sBz2UIwsw5S5zgGpSEwvg%3D" type="video/mp4">

</video>

</body>

</html>

If above program signed url is text file , then we can access as below.

HttpGet getreq = **new** HttpGet(URI.*create*("https://chalapathisvssr.s3.amazonaws.com/sample/s3.txt?AWSAccessKeyId=AKIAI2EBDM4A7M7G6RXQ&Expires=1512736039&Signature=2%2FyHDC4l0Ueo93yrGxfsKEDAEI4%3D"));

CloseableHttpClient httpclient = HttpClients.*createDefault*();

CloseableHttpResponse res = httpclient.execute(getreq);

InputStream is = res.getEntity().getContent();

String actual = IOUtils.*toString*(is);

System.***out***.println(actual);

Ref Url :

<http://docs.aws.amazon.com/AmazonS3/latest/dev/ShareObjectPreSignedURLJavaSDK.html>

<http://docs.aws.amazon.com/AmazonS3/latest/dev/PresignedUrlUploadObjectJavaSDK.html>

Generating Amazon S3 Pre-signed URLs with SSE

There are 4 types for SSE server-side Encryption , see all 5parts

<https://aws.amazon.com/blogs/developer/generating-amazon-s3-pre-signed-urls-with-sse-part-1/>