

Blogs

Told you, we love sharing!

[Home \(/\)](#) / [Blog \(http://www.tothenew.com/blog/\)](http://www.tothenew.com/blog/) / [Technology \(http://www.tothenew.com/blog/category/technology/\)](http://www.tothenew.com/blog/category/technology/) / [AWS \(http://www.tothenew.com/blog/category/technology/aws/\)](http://www.tothenew.com/blog/category/technology/aws/)

Category

ENTER EMAIL

Subscribe to Our Blog

Streaming Videos On Demand using Amazon Cloudfront and S3

AWS ([HTTP://WWW.TOTHENEW.COM/BLOG/CATEGORY/TECHNOLOGY/AWS-2/](http://www.tothenew.com/blog/category/technology/aws-2/)), DEVOPS

([HTTP://WWW.TOTHENEW.COM/BLOG/CATEGORY/TECHNOLOGY/DEVOPS-TECHNOLOGY/](http://www.tothenew.com/blog/category/technology/devops-technology/)), TECHNOLOGY

([HTTP://WWW.TOTHENEW.COM/BLOG/CATEGORY/TECHNOLOGY/](http://www.tothenew.com/blog/category/technology/))

11 / JAN / 2016 BY [MOHIT DAYAL GUPTA \(HTTP://WWW.TOTHENEW.COM/BLOG/AUTHOR/MOHIT-GUPTA-INTELLIGRAPE-COM/\)](http://www.tothenew.com/blog/author/mohit-gupta-intelligrape-com/) [3 COMMENTS \(HTTP://WWW.TOTHENEW.COM/BLOG/STREAMING-VIDEOS-ON-DEMAND-USING-AMAZON-CLOUDFRONT-AND-S3/#COMMENTS\)](http://www.tothenew.com/blog/streaming-videos-on-demand-using-amazon-cloudfront-and-s3/#comments)

Share this blog

Email

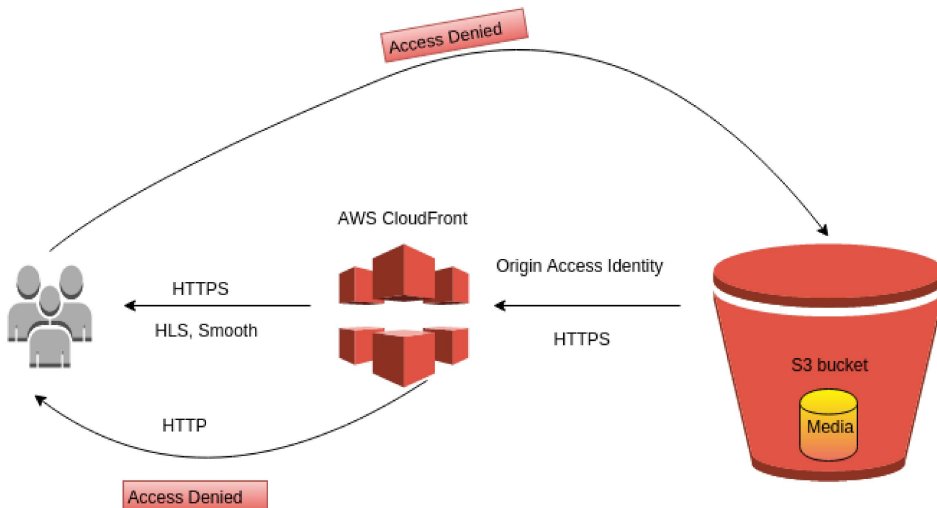
Twitter

Facebook

LinkedIn

Google+

I came across a scenario where we have to stream Videos On Demand (VOD) using Amazon CloudFront and Amazon Simple Storage Service (S3). The on-demand streaming is done using Cloudfront Content Delivery Network (CDN). The videos to be served are stored on Amazon S3. I have designed a secure architecture for the same setup.

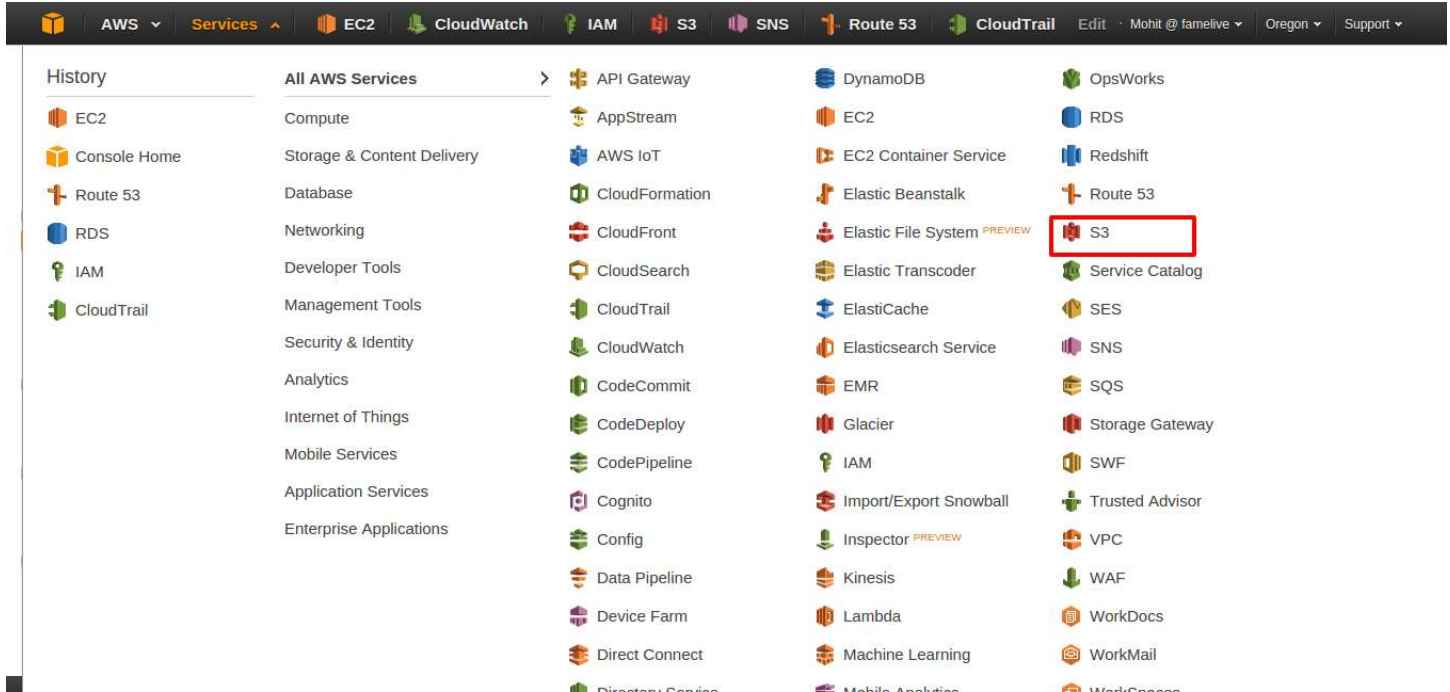


- The user sends the request to AWS CloudFront CDN for the video to be streamed.
- The request is preferred to be sent on the HTTPS protocol. For more secure streaming, all requests being made through HTTP protocol are denied by the CloudFront.
- The videos to be served are stored on AWS S3. The CORS configuration is set so as to allow access from the desired domains only.
- The access to S3 bucket is made through HTTPS protocol for more secure architecture. The HTTP requests are denied by AWS S3 policy which will be described later in the blog.
- The **CloudFront Origin Access Identity**, which is a special Cloudfront user is used to restrict access to the content in S3 buckets. The access is given only through the CloudFront so that the users cannot access the content directly by using S3 url.

Creating Amazon S3 bucket to store video content

The following steps are performed to create Amazon S3 bucket and modify its permissions:

1. Go to S3 console by logging into AWS console. Click on S3 option as shown below:



2. Create a new S3 bucket by clicking on **Create Bucket** button. Specify the name and region of the bucket.

3. Upload the video files to be served on the bucket by using AWS console or AWS CLI.

4. Create a file named **crossdomain.xml** and upload it. The file should contain the following content:

```
1 <cross-domain-policy>
2 <allow-access-from domain="*" secure="false"/>
3 <site-control permitted-cross-domain-policies="all"/>
4 </cross-domain-policy>
```

5. Update CORS information by clicking on **Edit CORS configuration** as shown:

Bucket: [redacted]
 Region: Singapore
 Creation Date: Mon Jan 04 12:03:20 GMT+530 2016
 Owner: [redacted]

▼ Permissions

You can control access to the bucket and its contents using access policies. [Learn more.](#)

Grantee: [redacted] ☒ List ☒ Upload/Delete ☒ View Permissions ☒ Edit Permissions X

[Add more permissions](#) [Edit bucket policy](#) [Edit CORS Configuration](#)

[Save](#) [Cancel](#)

► Static Website Hosting

► Logging

► Events

6. Put the following content in the CORS configuration file:

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
3  <CORSRule>
4  <AllowedOrigin>*/AllowedOrigin>
5  <AllowedMethod>PUT</AllowedMethod>
6  <AllowedMethod>POST</AllowedMethod>
7  <AllowedMethod>DELETE</AllowedMethod>
8  <AllowedMethod>GET</AllowedMethod>
9  <ExposeHeader>ETag</ExposeHeader>
10 <AllowedHeader>*</AllowedHeader>
11 </CORSRule>
12 </CORSConfiguration>

```

Here **AllowedOrigin** tag ensures the host from which the access to the content is allowed. You can set it to your desired host.
AllowedMethod tag ensures methods allowed on the bucket.

7. Now update the S3 bucket policy by clicking on **Add Bucket Policy** near **Edit CORS configuration** option. The policy must contain the following content:

```

1  {
2    "Version": "2008-10-17",
3    "Id": "some_policy",
4    "Statement": [
5      {
6        "Sid": "AddPerm",
7        "Effect": "Deny",
8        "Principal": "*",
9        "Action": "s3:*",
10       "Resource": "arn:aws:s3:::<strong>bucket_name</strong>/*",
11       "Condition": {
12         "Bool": {
13           "aws:SecureTransport": false
14         }
15       }
16     },
17     {
18       "Sid": "2",
19       "Effect": "Allow",
20       "Principal": {
21         "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity <strong>
22         <strong>OriginAccessIdentityID</strong></strong>"
23       },
24       "Action": "s3:GetObject",
25       "Resource": "arn:aws:s3:::<strong><strong>bucket_name</strong></strong>/*"
26     }
27   ]
28 }

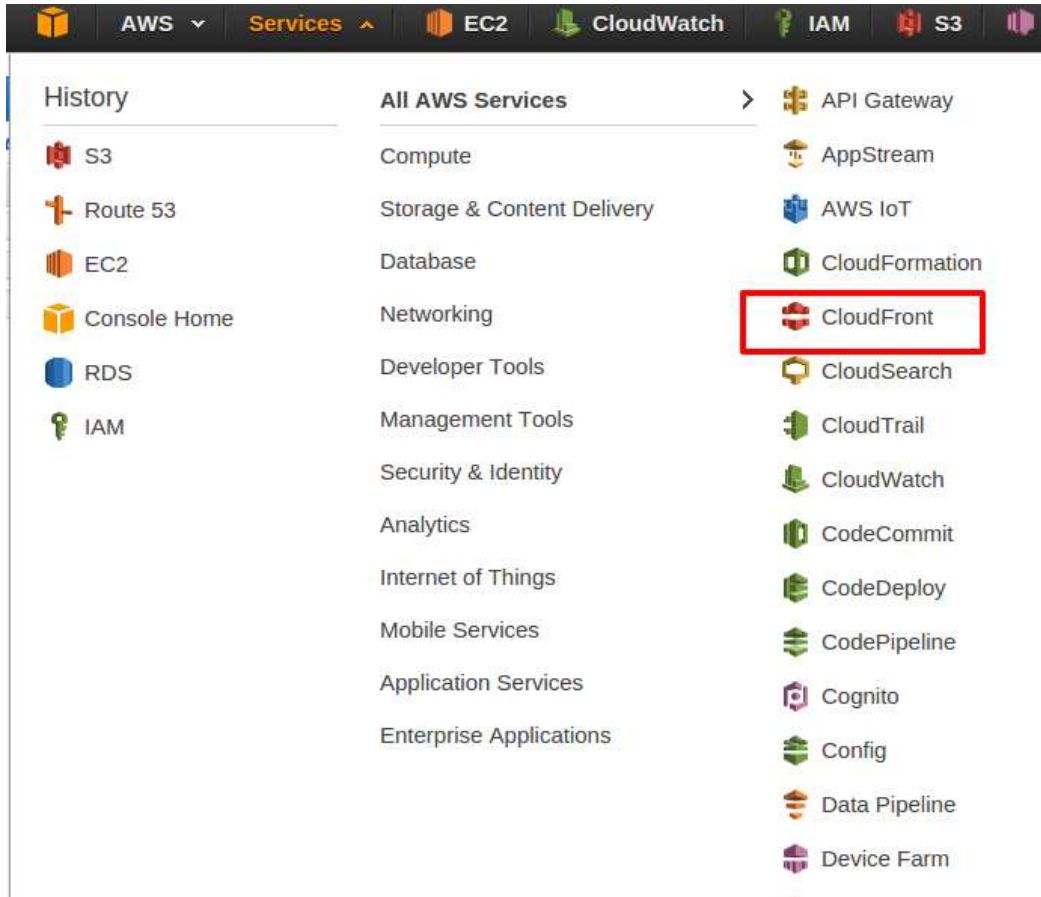
```

The **bucket_name** should be replaced by the name of the bucket and **OriginAccessIdentityID** should be replaced by CloudFront Origin Access Identity. It will be obtained when we create the CloudFront distribution below.

Here **"Effect": "Deny"** **aws:SecureTransport": false** is used to prevent HTTP access to S3 bucket. The GetObject permission is given of the content in S3 only to the cloud front user.

Creating Amazon CloudFront distribution to stream video content

1. Go to CloudFront console by logging into AWS console. Click on CloudFront option as shown below:



2. Click then on **Create Distribution** to create a new cloud front distribution. The new distribution will be used to stream videos.

3. Click on **Create Web Distribution**.

4. Now enter the required details in the boxes as shown in the image below:

Create Distribution ?

Origin Settings

Origin Domain Name	<input type="text"/>	i				
Origin Path	<input type="text"/>	i				
Origin ID	<input type="text"/>	i				
Origin Custom Headers	<table border="0"> <tr> <th>Header Name</th> <th>Value</th> </tr> <tr> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </table>	Header Name	Value	<input type="text"/>	<input type="text"/>	i
Header Name	Value					
<input type="text"/>	<input type="text"/>					

+

Default Cache Behavior Settings

Path Pattern	Default (*)	i
Viewer Protocol Policy	<input checked="" type="radio"/> HTTP and HTTPS <input type="radio"/> Redirect HTTP to HTTPS <input type="radio"/> HTTPS Only	i
Allowed HTTP Methods	<input checked="" type="radio"/> GET, HEAD <input type="radio"/> GET, HEAD, OPTIONS <input type="radio"/> GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE	i

Origin is where you store your content to be served.

- **Origin Domain Name:** Choose Amazon S3 bucket you created above.

- **Origin Path:** Leave it blank.
- **Origin ID:** It will be automatically set on choosing bucket in Origin Domain Name above.
- **Restrict Bucket Access:** Choose Yes.
- **Origin Access Identity:** Choose to create a new Identity.
- **Viewer Protocol Policy:** Choose HTTPS only.
- **Allowed HTTP methods:** Choose the methods you want to allow.

5. **SSL Certificate:** You can use default CloudFront certificate or can create your own certificate. To create your own certificate, follow the below steps:

- We need a **server certificate**, a **private key** and a **certificate chain** to create our own SSL certificate.
- Then using AWS CLI create certificate by:

```
1 | aws iam upload-server-certificate --server-certificate-name YOURNAME --certificate-body
   file://YOURSIGNEDCERT.pem --private-key file://YOURPRIVATEKEY.pem --certificate-chain
   file://INTERMEDIATECERT.pem --path /cloudfront/path/
```

YOURNAME – Identifying the certificate once it is uploaded to AWS console.

YOURSIGNEDCERT – The certificate received from Certificate Authority(CA).

YOURPRIVATEKEY – The private key, which is generated as a part of certificate process.

INTERMEDIATECERT – The intermediate key provided by CA.

The certificate will then be available on your AWS console. The certificate generated above can be verified by:

```
1 | aws iam get-server-certificate --server-certificate-name YOURNAME
```

Leave other options as default and click on **Create Distribution**.

6. On Web distribution page, you will get DNS of your CloudFront as *******.net**. Copy this and create a Cname in Route 53 as shown:

7. Now update this DNS in CloudFront distribution. Go to your web distribution. Click on General and then click on Edit. Then update DNS as shown:

Edit Distribution

Distribution Settings

8. The origin access identity will be obtained after you create the cloud front distribution. After creating the distribution, edit the distribution and origin access identity will be seen as below:

CloudFront Distributions > [Distribution ID](#) ?

General **Origins** Behaviors Error Pages Restrictions Invalidations

Create Origin Edit Delete

Origin Domain Name and Path	Origin ID	Origin Type	Origin Access Identity
<input type="checkbox"/> https://s3.amazonaws.com/...	...	S3 Origin	origin-access-identity/cloudfront/...

The CloudFront distribution will take some time to create. You can check the status at main distribution page. After the distribution is created, you can check out the complete secure architecture. The few testing steps are as follows:

- The content present on S3 will not directly be accessible by using the S3 url.
- Access the content by `https://<cloudfront_dns>/<folder_name>/<file_name>`.
- The content should not be accessible by `http://<cloudfront_dns>/<folder_name>/<file_name>`.

The secure architecture has been implemented. The architecture can be modified at ease by modifying the parameters or policies.

Tag -

AWS Cloudfront (<http://www.tothenew.com/blog/tag/aws-cloudfront/>) Aws S3 (<http://www.tothenew.com/blog/tag/aws-s3/>)

AWS Video Streaming (<http://www.tothenew.com/blog/tag/aws-video-streaming/>)

Media Streaming Cloudfront (<http://www.tothenew.com/blog/tag/media-streaming-cloudfront/>)

Secure Media Streaming (<http://www.tothenew.com/blog/tag/secure-media-streaming/>)

Serving VOD Content AWS (<http://www.tothenew.com/blog/tag/serving-vod-content-aws/>)

VOD Streaming Through AWS (<http://www.tothenew.com/blog/tag/vod-streaming-through-aws/>)

FOUND THIS USEFUL? SHARE IT





Whitepaper

DevOps Practices and Principles To Improve IT Efficiency

COMMENTS (3)



Download Whitepaper (<http://insights.tothenew.com/devops-practices-principles-for-it-efficiency>)
Deft Berlin -August 16, 2017 at 2:16 pm (<http://www.tothenew.com/blog/streaming-videos-on-demand-using-amazon-cloudfront-and-s3/comment-page-1/#comment-185463>)

This blog is awesome. Kudos and a big thanks to the person who have written it.

[Reply \(/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=185463#Respond\)](/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=185463#Respond)



Tarun Saxena -March 27, 2017 at 11:05 am (<http://www.tothenew.com/blog/streaming-videos-on-demand-using-amazon-cloudfront-and-s3/comment-page-1/#comment-178387>)

Amazing blog. Thanks for sharing this 😊

[Reply \(/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=178387#Respond\)](/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=178387#Respond)



Mohit Gupta -March 27, 2017 at 11:28 am (<http://www.tothenew.com/blog/streaming-videos-on-demand-using-amazon-cloudfront-and-s3/comment-page-1/#comment-178389>)

My pleasure 😊

[Reply \(/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=178389#Respond\)](/Blog/Streaming-Videos-On-Demand-Using-Amazon-Cloudfront-And-S3/?Replytoocom=178389#Respond)

LEAVE A COMMENT -

Name *

Email *