

---

# **Amazon CloudFront**

## **Getting Started Guide**

**API Version 2012-03-15**



## **Amazon CloudFront: Getting Started Guide**

Copyright © 2010 Amazon Web Services LLC or its affiliates. All rights reserved.

Getting Started with Amazon CloudFront ..... 1

Start Using Amazon CloudFront with Amazon S3 ..... 4

Where Do I Go from Here? ..... 10

Troubleshooting ..... 14

About This Guide ..... 17

# Getting Started with Amazon CloudFront

---

CloudFront is a content delivery service from Amazon Web Services that helps you improve the performance, reliability, and availability of your web sites and applications.

You can get started very quickly with CloudFront—there are only a few steps to follow.

1. Sign in to the AWS Management Console and open the Amazon CloudFront console at <https://console.aws.amazon.com/cloudfront/>.
2. In the CloudFront console, click Create Distribution.
3. Use the CloudFront domain name, or specify a CNAME to reference content in your web pages or applications.

The rest of this guide will take you through these steps in more detail, based on what you want to know about CloudFront. You can go on to either of the following topics.

- [What Is Amazon CloudFront and Why Do I Need It? \(p. 1\)](#) (below)
- [Start Using Amazon CloudFront with Amazon S3 \(p. 4\)](#)

## What Is Amazon CloudFront and Why Do I Need It?

Amazon CloudFront works by distributing your web content (such as images, video, and so on) using a network of edge locations around the world. Your content is served from your configured Amazon S3 bucket or custom origin, to the edge location that has the shortest latency for the user who requests it. (For more information on buckets and origin servers, see [Start Using Amazon CloudFront with Amazon S3 \(p. 4\)](#).)

This concept is best illustrated by an example. Suppose you were serving this simple URL from a traditional web server, not from CloudFront:

## Amazon CloudFront Getting Started Guide

### What Is Amazon CloudFront and Why Do I Need It?

[http://eoimages.gsfc.nasa.gov/images/imagerecords/57000/57723/globe\\_west\\_540.jpg](http://eoimages.gsfc.nasa.gov/images/imagerecords/57000/57723/globe_west_540.jpg). (This image is owned by NASA. The image comes from the Visible Earth website, <http://visibleearth.nasa.gov/>.)

Your end users can easily navigate to this URL and see the image that it returns. However, your end users might not be aware that their requests for this image were actually routed through a number of individual networks. Fundamentally, this is because the Internet is comprised of a complex collection of individual but interconnected networks. A request for data like this image is routed from one network to another until the data is found.

Suppose that your web server is in Seattle, WA, USA, and an end user makes a request to this URL from Austin, TX, USA. The traceroute list below (courtesy of [www.WatchMouse.com](http://www.WatchMouse.com)) shows how this request is routed.

1	vrid-225.core-sw.aus.us.siteprotect.com (216.139.225.1) 0.627 ms
2	xe-3-4.brdr-rtr-02.aus.us.siteprotect.com (216.139.253.53) 0.219 ms
3	66.113.197.121 0.452 ms
4	xe-5-2-0.edge3.Dallas1.Level3.net (4.59.112.37) 4.978 ms
5	ae-73-70.ebr3.Dallas1.Level3.net (4.69.145.116) 9.817 ms
6	ae-7-7.ebr3.Atlanta2.Level3.net (4.69.134.22) 30.570 ms
7	ae-2-2.ebr1.Washington1.Level3.net (4.69.132.86) 38.801 ms
8	ae-81-81.csw3.Washington1.Level3.net (4.69.134.138) 41.795 ms
9	ae-3-89.edge2.Washington1.Level3.net (4.68.17.145) 39.193 ms
10	72.21.222.139 35.767 ms



As you can see, the request was routed 10 times before the image was retrieved. This is not an unusually high number of hops, but it does illustrate how much work is needed to retrieve even a single image.

Continuing the example, consider the case when a request is made from Europe. Because the request is still being served from a traditional web server in a single location, the request must travel over even more networks. The number of networks and the distance it must travel has significant impact on the performance, reliability, and availability of the image.

This is where CloudFront can help. By using CloudFront to distribute your data, you can dramatically decrease the routing needed. If you served this image using CloudFront and your end users requested the same image from Europe, they would see the same object, but their request would take a very different path. The number of networks that the request uses is dramatically reduced. This is because CloudFront will detect where a request is being made from and service that request from a nearby edge location. This improves performance: end users get lower latency (the time it takes to load the first byte of the object) and higher data transfer rates. You also get increased reliability and availability because there is no longer a central point of failure—copies of your object are now held in edge locations around the world.

CloudFront improves performance in extreme cases, such as when a user makes a request from Europe to data stored originally in the United States. However, CloudFront works just as well at more granular levels.

Go to [Start Using Amazon CloudFront with Amazon S3 \(p. 4\)](#).

# Start Using Amazon CloudFront with Amazon S3

---

This section can help you to quickly get started using CloudFront with Amazon S3.



## Note

This exercise describes how to use CloudFront with Amazon S3 origins. (An origin is the server location where you store the original, definitive version of the objects you want to distribute.) For more information about origin servers, go to *The Origin Server* in [Parts of a Distribution](#). To learn how to create a distribution with a custom origin, go to [Creating a Distribution with a Custom Origin](#) in the *Amazon CloudFront Developer Guide*.

There are only a few basic steps to start delivering your content using CloudFront. The first step is signing up. After that, you create an CloudFront distribution, and then use the CloudFront domain name to reference content in your web pages or applications.

## Step One: Sign up for Amazon Web Services

If you have not already done so, sign up for Amazon Web Services at <http://aws.amazon.com>. Just click **Sign Up Now** and enter any required information.



## Step Two: Upload your content to Amazon Simple Storage Service (S3)

If you have already uploaded your content to Amazon S3, then you can skip this step and proceed to [Step Three: Create a CloudFront Distribution \(p. 6\)](#).

## Amazon CloudFront Getting Started Guide

### Step Two: Upload your content to Amazon Simple Storage Service (S3)

---

If you are new to Amazon S3, you must sign up for Amazon S3 because the content you deliver with CloudFront will be stored in an Amazon S3 bucket.

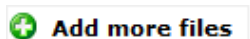
Content can consist of just about any type of object, including images and videos. A bucket is a container for your objects and is similar to a server directory folder. Buckets can contain objects or folders, but you cannot nest buckets. The bucket resides on a server referred to as your origin server. You can create multiple buckets, and there is no limit to the amount of data you can store on Amazon S3.

#### To upload your content to Amazon S3

1. Sign in to the AWS Management Console and open the Amazon S3 console at <https://console.aws.amazon.com/s3/>.
2. In the Amazon S3 console, click **Create Bucket**.
3. In the **Create Bucket** dialog:
  - a. Enter a bucket name.

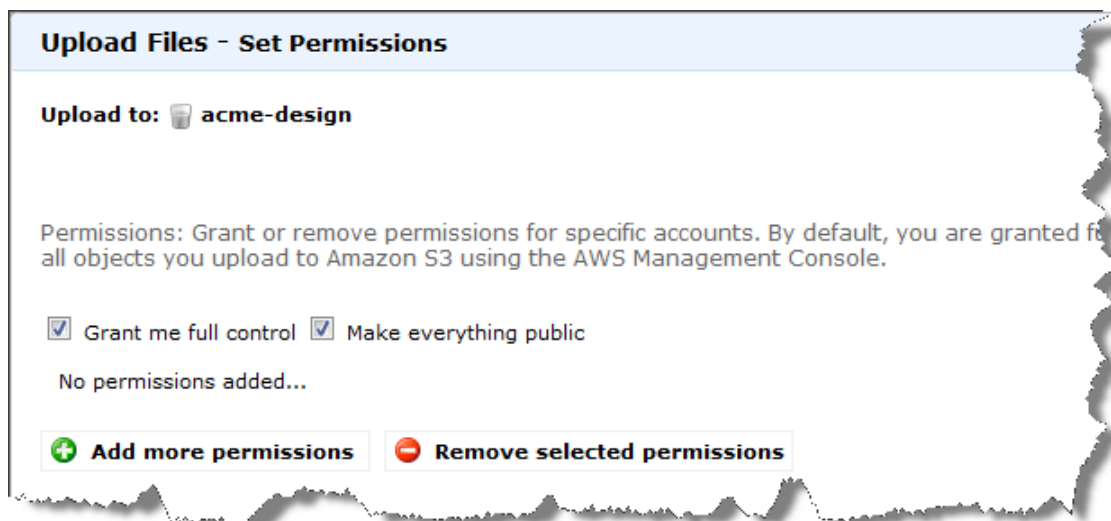
Amazon S3 requires that bucket names are globally unique, just like a domain name: no two buckets can have the same name, even if they're owned by different customers. For example, you could create a bucket name by combining the name of your company with the name of your company project, such as example-myawsbucket. But you can choose any name you want as long as it's not already in use by another bucket.
  - b. Select a region for your bucket. By default, Amazon S3 creates buckets in the US-Standard region. You should choose a region close to you to optimize latency, minimize costs, or to address regulatory requirements.
4. Click **Create**.

Amazon S3 generates an URL for your bucket. That URL is based on the name of your bucket. Using the example from the previous step, the URL would be: `http://s3.amazonaws.com/acme-design`. Note that this URL is not the same URL you will use to distribute your content through CloudFront. Once your bucket has been created, your next step is to upload your content and set the object properties.
5. Make sure the bucket you just created is selected, then click **Upload**.
6. Click **Add More Files**, then choose the file(s) you want to upload.



7. Click **Set Permissions**, then select **Make Everything Public**.





The easiest way to use CloudFront with Amazon S3 is to make all your objects publicly readable in Amazon S3. To do this, you must explicitly enable public read privileges for each object you upload to Amazon S3. (This is one of the most common mistakes when working with CloudFront and Amazon S3. You must explicitly grant privileges to each object in Amazon S3.)

If you want to restrict who can download your content, you can enable the CloudFront private content feature. For more information about distributing private content, go to [Using a Signed URL to Serve Private Content](#) in the *Amazon CloudFront Developer Guide*.

8. Click **Start Upload**.

When the upload completes, you can navigate to this item by its URL. In the case of the previous example, the URL would be: `http://s3.amazonaws.com/example-myawsbucket/[file name]`. Use your S3 URL to verify that your content is publicly accessible, but remember that this is not the URL you will use when you are ready to distribute your content.

## Step Three: Create a CloudFront Distribution

### To create a CloudFront distribution

1. Open the Amazon CloudFront console at <https://console.aws.amazon.com/cloudfront/>.
2. Click **Create Distribution**.



3. In the **Create Distribution** dialog:
  - a. Select **Streaming** or **Download** to indicate the distribution delivery method.  
This document focuses on download distributions. For more information about streaming distributions, go to [Streaming Media on Demand](#) in the *Amazon CloudFront Developer Guide*.
  - b. Select your Amazon S3 bucket from the **Origin** drop-down list. (If you do not already use Amazon S3, you must complete the steps described in [Step Two: Upload your content to Amazon Simple Storage Service \(S3\)](#) (p. 4) before creating a distribution).



### Important

If CloudFront cannot read your content in Amazon S3, CloudFront cannot access it. You must ensure that your object permissions are set to **Make Everything Public** for each object in your Amazon S3 bucket. If you are distributing private content, you need to create a CloudFront origin access identity so that CloudFront can access your Amazon S3 content. For more information, go to [Creating a CloudFront Origin Access Identity](#) in the *Amazon CloudFront Developer Guide*.

- c. Select **On** if you want to enable access logging. You can optionally enable this functionality later. For more information about access logs, go to [Access Logs](#) in the *Amazon CloudFront Developer Guide*.
- d. If you are using a CNAME with your distribution, enter your CNAME. A CNAME is useful because it lets you use your domain name for your links instead of the domain name CloudFront provides you.

CloudFront automatically generates a URL for your distribution, but that URL can be cumbersome to work with. An example of a generated URL is `d604721fxaaqy9.cloudfront.net`. A CNAME, or canonical name, is an entry in a DNS table that maps a friendly name to the actual CloudFront distribution name.

For example, if you owned `www.example.com` and your CloudFront URL was `d604721fxaaqy9.cloudfront.net`, you could use a CNAME that would specify `www.example.com` as an alias for `d604721fxaaqy9.cloudfront.net`. If you are comfortable using a URL such as `d604721fxaaqy9.cloudfront.net` in your web page or application, there is no need to use a CNAME. Creating a CNAME is an optional step.



### Important

You must be the owner of any domain name that you specify as an alias of your distribution's domain name.

In addition to entering your CNAME in this dialog, you will need to create a DNS entry for your CNAME. CNAMEs are managed by whoever manages the DNS entries for your site. This is usually your web hosting provider. The exact procedure for configuring DNS depends on your DNS server or DNS provider, but generally involves creating a CNAME record in the DNS system to establish the alias between your domain name and the CloudFront domain name for your distribution. Usually, a web hosting provider will discuss how to alter your DNS entries in their support documentation.

New DNS entries usually take a few minutes to propagate. When it does, you can access your content at its CNAME. This is the base URL that you will use to access your content through CloudFront.

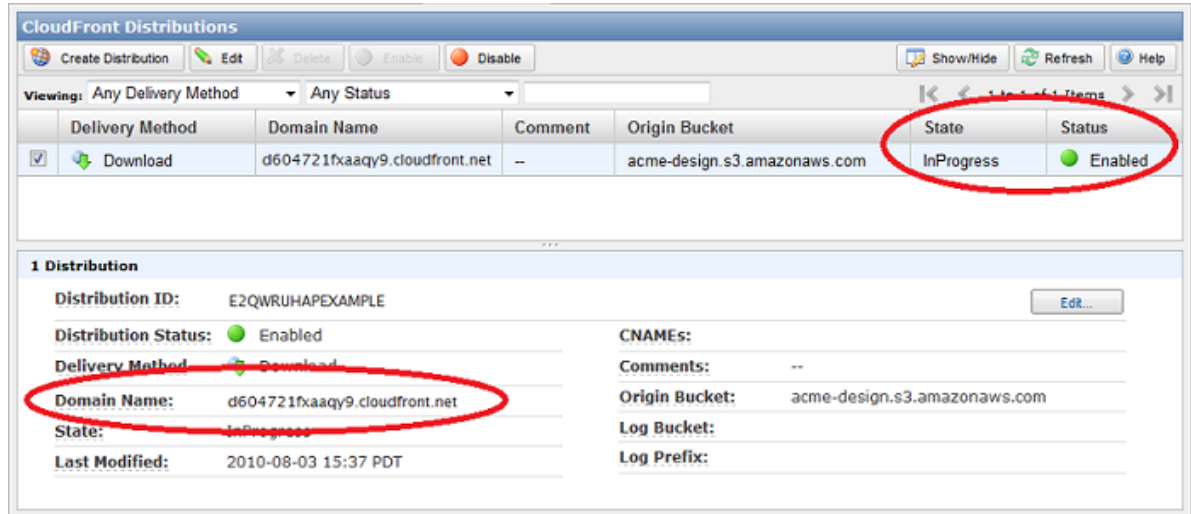
- e. Enter any comments you want to record with regard to this distribution. Comments you enter are for your reference only.
- f. Select **Enabled** to enable your distribution for access upon its creation, or **Disabled** if you want to wait and to make it accessible later.

- 4. Click **Create**. Your distribution is created.

After creating the distribution, it might take a few minutes for the distribution to deploy. You can monitor the progress in the State field.

## Amazon CloudFront Getting Started Guide

### Step Four: Use the CloudFront domain name to reference content in your web pages or applications



The screenshot shows the 'CloudFront Distributions' console. At the top, there are buttons for 'Create Distribution', 'Edit', 'Delete', 'Enable', and 'Disable'. Below these are filters for 'Viewing: Any Delivery Method' and 'Any Status'. A table lists the distributions. The first distribution is highlighted with a red circle around its 'State' and 'Status' columns, which show 'InProgress' and 'Enabled' respectively. Below the table, the details for the selected distribution are shown. The 'Domain Name' field is circled in red and shows 'd604721fxaaqy9.cloudfront.net'. Other fields include 'Distribution ID', 'Delivery Method', 'State', 'Last Modified', 'CNAMEs', 'Comments', 'Origin Bucket', 'Log Bucket', and 'Log Prefix'.

Delivery Method	Domain Name	Comment	Origin Bucket	State	Status
Download	d604721fxaaqy9.cloudfront.net	--	acme-design.s3.amazonaws.com	InProgress	Enabled

**1 Distribution**

**Distribution ID:** E2QWRUHAPEXAMPLE

**Distribution Status:** Enabled

**Delivery Method:** Download

**Domain Name:** d604721fxaaqy9.cloudfront.net

**State:** InProgress

**Last Modified:** 2010-08-03 15:37 PDT

**CNAMEs:**

**Comments:** --

**Origin Bucket:** acme-design.s3.amazonaws.com

**Log Bucket:**

**Log Prefix:**

When the status becomes Enabled and the state changes from InProgress to Deployed, your distribution is ready. Also, as you can see above, the automatically generated domain name for your distribution is visible. If you chose to use a CNAME, it will also be visible. It will generally take less than 15 minutes for a distribution to deploy.



#### Note

You will only get the benefits of using CloudFront if you use the cloudfront.net domain name (or your CNAME pointing to it) in your applications or web pages.

You are ready to reference your content with your new CloudFront URL or CNAME.

## Step Four: Use the CloudFront domain name to reference content in your web pages or applications

After you've created your distribution, CloudFront knows where your Amazon S3 origin server is, and you know the domain name associated with the distribution. You can create a link to your Amazon S3 bucket content with that domain name and have CloudFront serve it.



#### Note

You must wait until the distribution's state is Deployed before testing your links.

#### To link to your object

1. Copy the following HTML into a new file and replace the domain name with your distribution's domain name and object name with the name of your content.

```
<html>
<head>My CloudFront Test</head>
```

## Amazon CloudFront Getting Started Guide

### Step Four: Use the CloudFront domain name to reference content in your web pages or applications

---

```
<body>
<p>My text content goes here.</p>
<p> 
</body>
</html>
```

For example, if your domain name was `d604721fxaaqy9.cloudfront.net` and your object was `image.jpg`, the URL for the link would be:

`http://d604721fxaaqy9.cloudfront.net/image.jpg`.

If your object is in a folder within your bucket, then the folder must also be included in the URL. For example, if `image.jpg` was located in your bucket's `Images` folder, then the URL would be:  
`http://d604721fxaaqy9.cloudfront.net/images/image.jpg`.

2. If you are creating a streaming distribution, you need to use an RTMP protocol instead of HTTP, and you must configure your playback client. For more information, go to [Streaming Media on Demand](#) in the *Amazon CloudFront Developer Guide*.
3. Save these changes to your file.
4. Open your web page in a browser to ensure you can see your image. If you cannot see your image, try these steps again, or read the tips in [Troubleshooting \(p. 14\)](#).

The browser returns your page with the embedded image file, served from the edge location that CloudFront determined was appropriate to serve the object.

Congratulations! You've used CloudFront for the first time to serve an image.

For more information on using CloudFront, go to [Where Do I Go from Here? \(p. 10\)](#)

## Where Do I Go from Here?

---

Although fairly simple to use, CloudFront is rich in functionality. There are a variety of resources you can use to learn more about CloudFront. In addition to the [Amazon CloudFront documentation](#), resources include AWS SDKs, feature summaries and links to third-party applications that work with CloudFront. The list below includes a few of these resources that you might find helpful.

### Topics

- [Amazon CloudFront SDKs](#) (p. 10)
- [Amazon CloudFront Getting Started Video](#) (p. 10)
- [Using Amazon CloudFront Logging](#) (p. 11)
- [Setting a Default Root Object](#) (p. 11)
- [Invalidating Objects](#) (p. 11)
- [Distributing Streaming Media](#) (p. 11)
- [Changing the Expiration Time \(TTL\) of Your Content](#) (p. 12)
- [Increasing Web Site Performance](#) (p. 12)
- [Creating Secure Connections Using HTTPS](#) (p. 12)
- [Distributing Private Content](#) (p. 12)
- [Using Custom Origins](#) (p. 12)
- [Amazon CloudFront Third-party Tools Overview](#) (p. 12)
- [Using CloudFront with a Content Management System](#) (p. 12)

## Amazon CloudFront SDKs

AWS offers SDKs that let you access CloudFront programmatically.

- [AWS SDK for .NET](#)
- [AWS SDK for PHP](#)

## Amazon CloudFront Getting Started Video

AWS: [Amazon CloudFront Getting Started](#)

## Using Amazon CloudFront Logging

AWS Blog: [Amazon CloudFront Request Logging](#) (for content delivered via HTTP)

AWS Blog: [Amazon CloudFront Now Supports Streaming Access Logs](#) (for content delivered via RTMP)

AWS Blog: [Enhanced CloudFront Logs, Now With Query Strings](#)

## Setting a Default Root Object

CloudBerry Lab: [How to set CloudFront Default Object with CloudBerry S3 Explorer](#)

## Invalidating Objects

For more information about third-party tools that support object invalidation, see "Third-Party Tools for Invalidating Objects" in the "[Object Invalidation](#)" section of the "Working With Objects" chapter of the *Amazon CloudFront Developer Guide*.

## Distributing Streaming Media

AWS Articles & Tutorials:

- [Configuring Amazon CloudFront Streaming Using Adobe's Flash Builder](#)
- [Configuring Amazon CloudFront Streaming Using Flowplayer](#)
- [Configuring Amazon CloudFront Streaming Using JW Player](#)

StreamingMedia.com: [How To Get Started With Amazon CloudFront Streaming](#)

loncannon.net:

- [iPhone Windowed HTTP Live Streaming Using Amazon S3 and Cloudfront Proof of Concept](#)
- [HTTP Live Video Stream Segmenter and Distributor](#)
- [iPhone Windowed HTTP Live Streaming Server](#)

Flowplayer.org: [Bandwidth detection: Make sure you reach your entire audience with good quality](#)

LongtailVideo.com (JW Player):

- [Using Amazon Web Services](#)
- [Using CloudFront](#)

## Changing the Expiration Time (TTL) of Your Content

AWS Blog: [Amazon CloudFront Object TTL Slashed!](#)

## Increasing Web Site Performance

AWS Blog: [Improving Global Application Performance](#)

## Creating Secure Connections Using HTTPS

AWS Blog: [Amazon CloudFront: HTTPS Access, Another Edge Location, Price Reduction](#)

## Distributing Private Content

AWS Blog: [New Amazon CloudFront Feature: Private Content](#)

## Using Custom Origins

AWS Blog: [New Amazon CloudFront Feature: Custom Origins](#)

## Amazon CloudFront Third-party Tools Overview

AWS Blog: [CloudFront Management Tool Roundup](#)

## Using CloudFront with a Content Management System

### Drupal

- Drupal.org: [CloudFront Installation](#)
- DrupalModules.com: [CloudFront Drupal Module](#)

### Joomla

- NickTemple.com: [CloudFront Plugin For Joomla!](#)

### Sitecore

- NTT Data Advisory Service: [AWS CloudFront Sitecore Integration](#)

## **WordPress**

- om4.com: [Using Amazon CloudFront with WordPress and WordPress MU](#)
- WordPress.org: [W3 Total Cache](#)
- WordPress.org: [Simple Amazon S3 Upload Form](#)
- WordPress.org: [OSSDL CDN Off-linker](#)
- WordPress.org: [My CDN](#)
- Inquisiter.com: [Amazon CloudFront CDN with a WordPress Blog](#)



# Troubleshooting

---

If you cannot view your CloudFront distribution, the following topics describe some common solutions.

## Topics

- [Did you sign up for both CloudFront and Amazon S3? \(p. 14\)](#)
- [Are your Amazon S3 bucket and object permissions set correctly? \(p. 14\)](#)
- [Is your CNAME correctly configured? \(p. 15\)](#)
- [Are you referencing the correct URL for your CloudFront distribution? \(p. 15\)](#)
- [If you are working with a streaming distribution, are your URL and your playback client correctly configured? \(p. 16\)](#)
- [Do you need help troubleshooting a custom origin? \(p. 16\)](#)

## Did you sign up for both CloudFront and Amazon S3?

To use Amazon CloudFront with an Amazon S3 origin, you must sign up for both CloudFront and Amazon S3, separately. For more information about signing up for CloudFront and Amazon S3, see [Start Using Amazon CloudFront with Amazon S3 \(p. 4\)](#).

## Are your Amazon S3 bucket and object permissions set correctly?

If you are using CloudFront with an Amazon S3 origin, the original versions of your content are stored in an Amazon S3 bucket. The easiest way to use CloudFront with Amazon S3 is to make all your objects publicly readable in Amazon S3. To do this, you must explicitly enable public read privileges for each object you upload to Amazon S3.

If your content is not publicly readable, you need to create a CloudFront origin access identity so CloudFront can access it. To learn how to create a CloudFront origin access identity, see the description of creating a CloudFront origin access identity [Amazon CloudFront Developer Guide](#).

You should be aware that object properties and bucket properties are independent. You must explicitly grant privileges to each object in Amazon S3. Objects do not inherit properties from buckets and object properties must be set independently of the bucket.

## Is your CNAME correctly configured?

If you already have an existing CNAME record for your domain name, make sure you update that record or replace it with a new one that points to your distribution's domain name.

Also, make sure your CNAME record points to your distribution's domain name, not your Amazon S3 bucket. You can confirm that the CNAME record in your DNS system points to your distribution's domain name. To do so, use a DNS tool like dig (for information about dig, go to <http://www.kloth.net/services/dig.php>). Using dig, the CNAME record for your domain name is set up correctly if the name on the right side of CNAME in that line is your CloudFront distribution's domain name. If it's your Amazon S3 origin server bucket or some other domain name, then the CNAME record is set up incorrectly.

The following shows an example dig request on a domain name called images.yourdomain.com, and the relevant part of the response.

```
[prompt]> dig images.yourdomain.com

; <<> DiG 9.3.3rc2 <<> images.yourdomain.com
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15917
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 2, ADDITIONAL: 0
;; QUESTION SECTION:
;images.yourdomain.com.      IN      A
;; ANSWER SECTION:
images.yourdomain.com. 10800 IN  CNAME  d604721fxaaqy9.cloudfront.net.
...
...
```

You can find more details about CNAMEs in the [Amazon CloudFront Developer Guide](#).

## Are you referencing the correct URL for your CloudFront distribution?

Make sure the URL you are referencing uses your CloudFront distribution domain name (or your CNAME), not your Amazon S3 bucket or custom origin.

## If you are working with a streaming distribution, are your URL and your playback client correctly configured?

Streaming distributions require you to use an RTMP protocol instead of HTTP, and you must make a few minor configuration changes to your playback client. Information on creating streaming distributions can be found in the [Amazon CloudFront Developer Guide](#).

## Do you need help troubleshooting a custom origin?

If you need Amazon to help you troubleshoot a custom origin, we will probably need to inspect the `X-Amz-Cf-Id` header entries from your requests. If you are not already logging these entries, you might want to consider it for the future. For more information, go to [Custom Origins Best Practices](#) in the *Amazon CloudFront Developer Guide*.

# About This Guide

---

This is the *Amazon CloudFront Getting Started Guide*. It was last updated on 19 March 2012.

Amazon CloudFront is often referred to within this guide as "CloudFront." All copyrights and legal protections still apply.