

Working with AudienceStore and EventStore

This article shows how to access your AudienceStore and EventStore services, which store your unstructured audience and event data as compressed JSON files on Tealium's Amazon S3 bucket.

These services must be activated for your account. Contact your account manager for more info.

This article covers:

- [Prerequisites](#)
- [EventStore/AudienceStore Console](#)
 - [Viewing Files in the Console](#)
 - [Amazon S3 Access Key](#)
- [FTP Clients with Amazon S3 Support](#)
- [Amazon Command Line Interface](#)
 - [List Objects in S3](#)
 - [List All Objects in Root Folder:](#)
 - [List All Objects in Events Folder:](#)
 - [List All Objects in Specific Folder:](#)
 - [Get Single Object](#)
 - [Synchronize Local and Remote Folders](#)

Prerequisites

- [DataAccess](#)
- [Active AudienceStore Connector](#) (see the [Setup Guide](#))

– OR –

- [Active EventStream Feeds](#) (see the [Event Feed Setup Guide](#))

EventStore/AudienceStore Console

Viewing Files in the Console

You can browse the data files associated with each audience and feed via the [DataAccess](#) console. This is an easy way to verify that data is flowing through the system and to download a sample file to get familiar with the format.

To access files via the [DataAccess](#):

1. Navigate to [DataAccess > EventStore](#) or [DataAccess > AudienceStore](#).
2. Select the number of weeks of data to display.
3. Select the name of the Event Feed or the AudienceStore Action.
4. Click [Reload](#).
5. Click a date to expand the list of file details.
6. Find the file you want and click [Download](#).

The .gzip file will be saved to your computer where you can use an unzip utility to open the file.

Amazon S3 Access Key

Your data files can also be accessed via third-party tools such as an FTP client or the Amazon S3 command line interface. To allow these tools access to your files you will need the connection credentials for your bucket.

To get the Amazon S3 Access Key:

1. Navigate to [DataAccess > EventStore](#) or [DataAccess > AudienceStore](#).
2. Click [Get Amazon Access Key](#). The following fields are displayed:

- Access Key ID
- Secret Access Key
- Path

For security purposes the Secret Access Key is only displayed once, so it's important to store it securely for later use. If you ever lose this value you can regenerate a new one, but it will invalidate all previous connections that used the old value.

FTP Clients with Amazon S3 Support

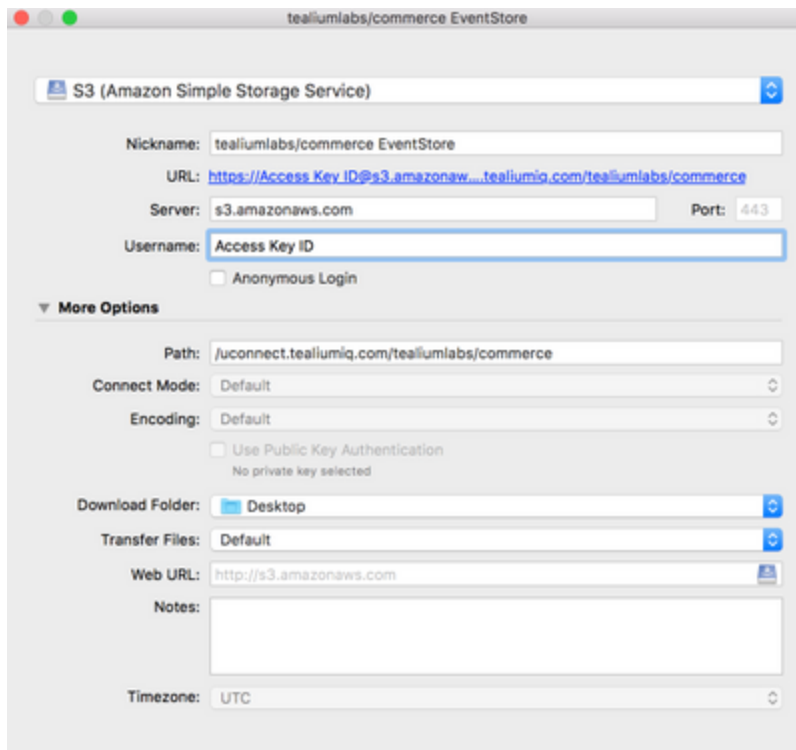
We recommend using a desktop application for a more convenient method of downloading a large number of AudienceStore/EventStore files.

Here are some client applications that work with Amazon S3.

- Windows: Cyberduck, CrossFTP
- Mac: Cyberduck, CrossFTP, Transmit

The primary benefit of using a GUI-based FTP client with S3 support is that you can point-and-click on individual files and folders to download from Amazon S3.

Below is a sample screenshot in Cyberduck for how to configure the connection. Note that there is not an input for the Secret Access Key within the configuration wizard, you will instead be prompted for it upon a connection attempt where it will be saved for future use.



Amazon Command Line Interface

For more technical users, the Amazon Command Line Interface (CLI) can be installed to give you full control over accessing your data files. The primary benefit of using a Amazon CLI is the ability to customize for your specific needs, such as syncing and automating the file retrieval from Amazon S3.

Some example uses:

- Initial bulk download of all historical log files
- Schedule hourly incremental download to grab only the newest generated log files
- Synchronize a local folder on your desktop or server to a remote folder on S3 so that they contain exactly the same log file content
- Download files before and/or after a certain LastModified date

To install the Amazon CLI, follow the instructions at:

<http://docs.aws.amazon.com/cli/latest/userguide/installing.html>

You will be prompted for your Access Key and Access Key ID when you call `aws configure` (you can leave Region Name and Output Format blank).

Once you configure CLI, you can now make queries using the `s3api` method, which we will use in the remaining examples.

List Objects in S3

The "list-objects" method allows you to list objects in your S3 directory. This is needed get the "key" for each object to download individual files.

List All Objects in Root Folder:

```
aws s3api list-objects --bucket subdomain.tealiumiq.com \
  --prefix {account}/{profile}/
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.

List All Objects in Events Folder:

```
aws s3api list-objects --bucket subdomain.tealiumiq.com \ --prefix {account}/{profile}/events/
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.

List All Objects in Specific Folder:

```
aws s3api list-objects --bucket subdomain.tealiumiq.com \ --prefix {account}/{profile}/events/{stream}/
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.

Get Single Object

The "get-object" method will download one specific remote key to a local location on your desktop or server.

```
aws s3api get-object --bucket subdomain.tealiumiq.com \
  --key {account}/{profile}/events/{stream}/{filename}.gz ./
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.

The "--key" component is made up of this format:

```
{account}/{profile}/events/{stream}/{filename}.gz
```

Synchronize Local and Remote Folders

The "sync" method takes a remote folder on Amazon S3 and synchronizes it with a local folder on your desktop or server. In this example we synchronize a specific remote Stream folder to a local folder on the desktop.

The "--dryrun" argument shows you what files would actually sync, without actually doing the download. To execute the actual download, remove the "--dryrun" argument.

```
aws s3 sync s3://subdomain.tealiumiq.com/{account}/{profile}/events// \
  ~/Desktop/temp --dryrun
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.

Lastly, you can also filter the "sync" method to only download files matching a specific filter. In this example, only the files that match the wildcard filter of `"*2015.06.14"` will be downloaded.

```
aws s3 sync s3://subdomain.tealiumiq.com/{account}/{profile}/events// \
  ~/Desktop/temp --exclude "*" --include "*2015.06.14*" --dryrun
```

Where the variable *subdomain* is the subdomain name, such as `dataaccess-us-east-1`.