

# Amazon S3 Import Integration

[Learn more about Amazon S3 Export Integration.](#)

The data connector for Amazon S3 enables you to import the data from your JSON, TSV, and CSV files stored in an S3 bucket.

For sample workflows on importing data from files stored in an S3 bucket, go to the [Treasure Box on Github](#).

An update to provide support for AssumeRole is coming in Spring 2022.

- [Prerequisites](#)
- [Use the TD Console to Create Your Connection](#)
  - [Create a New Connection](#)
  - [Transfer Your AWS S3 Data to Treasure Data](#)
    - [Connection](#)
    - [Source Table](#)
    - [Data Settings](#)
    - [Filters](#)
    - [Data Preview](#)
    - [Data Placement](#)
- [Validating Your Data Connector Jobs](#)
  - [How do I troubleshoot data import problems?](#)
  - [What can I do if the data connector for S3 job is running for a long time?](#)
- [Sample Workflow](#)

## Prerequisites

You must have basic knowledge of Treasure Data.

You must set up an access route in AWS if you are using an AWS S3 bucket located in the same region as your TD region. You set up the access route by specifying the VPC. For example, if in the US region, configure access through vpc-df7066ba. If in the Tokyo region, configure access through vpc-e630c182 and, for the EU01 region, vpc-f54e6a9e.

Look up the region of TD Console by the URL you are logging in to TD, then refer to the data connector of your region in the URL.

Region of TD Console	URL
US	<a href="https://console.treasuredata.com">https://console.treasuredata.com</a>
Tokyo	<a href="https://console.treasuredata.co.jp">https://console.treasuredata.co.jp</a>
EU01	<a href="https://console.eu01.treasuredata.com">https://console.eu01.treasuredata.com</a>

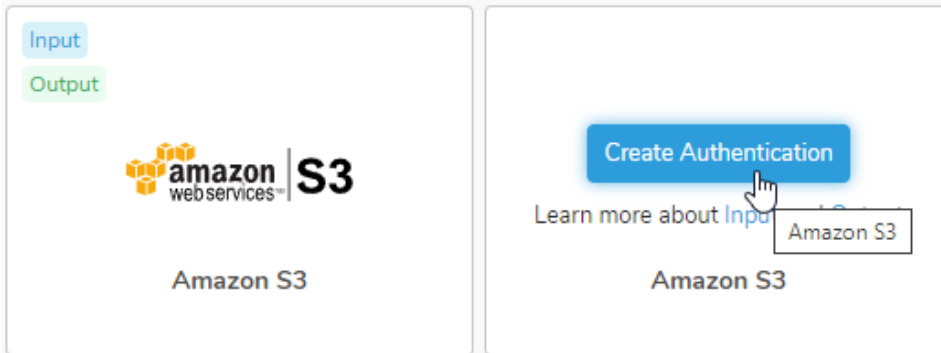
## Use the TD Console to Create Your Connection

You can use TD Console to create your data connector.

### Create a New Connection

When you configure a data connection, you provide authentication to access the integration. In Treasure Data, you configure the authentication and then specify the source information.

1. Navigate to **Integrations Hub > Catalog** and search for AWS S3.
2. Select **Create Authentication**.



3. New Authentication dialog opens. You need a Access key ID and a Secret access key to authenticate using credentials.

4. Set the following parameters. Select **Continue**. Name your new AWS S3 connection. Select **Done**.






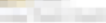


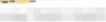
<b>Endpoint</b>	<ul style="list-style-type: none"> <li>S3 endpoint login user name. You can find region and endpoint information from <a href="#">AWS Document</a>. (Ex. <code>s3.ap-northeast-1.amazonaws.com</code>)</li> </ul>
<b>Authentication Method</b>	
<b>basic</b>	<ul style="list-style-type: none"> <li>Uses <code>access_key_id</code> and <code>secret_access_key</code> to authenticate. See <a href="#">AWS Programmatic access</a>. <ul style="list-style-type: none"> <li>Access Key ID</li> <li>Secret access key</li> </ul> </li> </ul>
<b>anonymous</b>	<ul style="list-style-type: none"> <li>Uses anonymous access. This auth method can access only public files.</li> </ul>
<b>session (Recommended)</b>	<ul style="list-style-type: none"> <li>Uses temporary-generated <code>access_key_id</code>, <code>secret_access_key</code> and <code>session_token</code>. (This authentication method is only available with data import. This can't be used with data export for now.) <ul style="list-style-type: none"> <li>Access Key ID</li> <li>Secret access key</li> <li>Secret token</li> </ul> </li> </ul>
<b>Access Key ID</b>	AWS S3 issued
<b>Secret Access Key</b>	AWS S3 issued

# Transfer Your AWS S3 Data to Treasure Data

After creating the authenticated connection, you are automatically taken to Authentications.

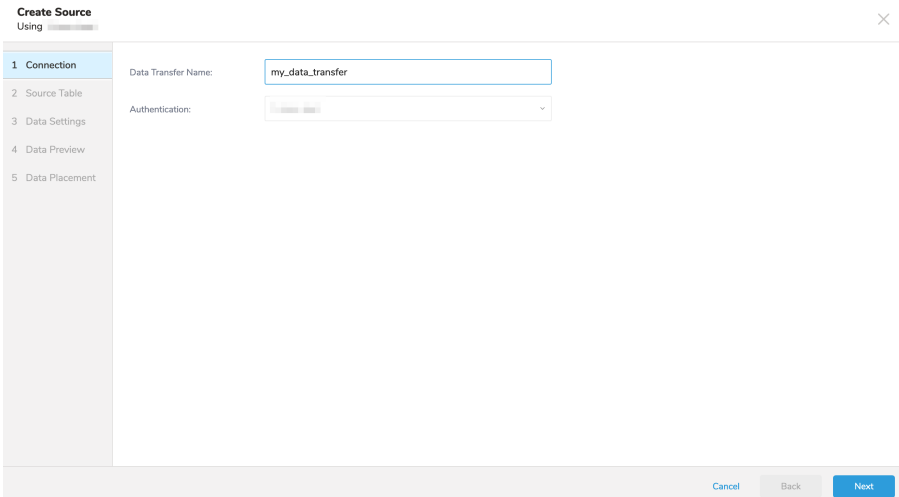
1. Search for the connection you created.
2. Select **New Source**.




Type	Authentication ^	Sources	
 AWS S3	by  	7	<a href="#">NEW SOURCE</a> <span>⋮</span>
 AWS S3	by  	0	<a href="#">NEW SOURCE</a> <span>⋮</span>
 AWS S3	by  	1	<a href="#">NEW SOURCE</a> <span>⋮</span>


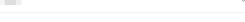
## Connection

1. Type a name for your **Source** in the Data Transfer field.
2. Click **Next**.



**Create Source**  
Using 

1 **Connection** | Data Transfer Name:

2 Source Table | Authentication:  

3 Data Settings

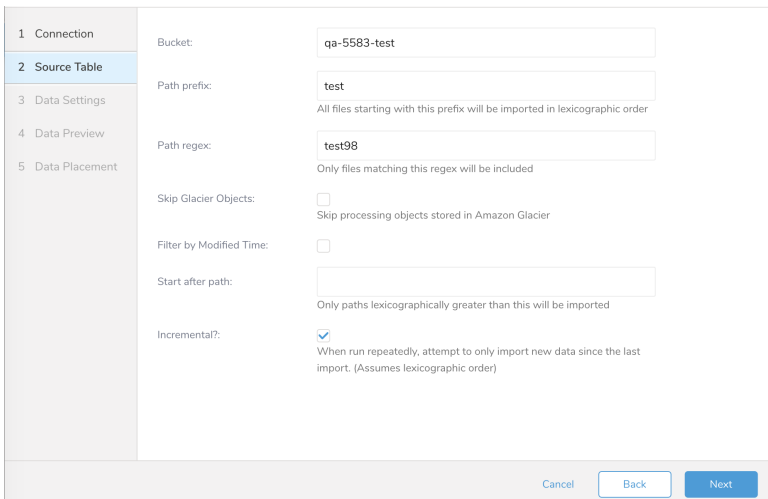
4 Data Preview

5 Data Placement

[Cancel](#) [Back](#) [Next](#)

## Source Table

1. The Source dialog opens. Edit the following parameters



1 Connection

2 **Source Table** | Bucket:

3 Data Settings | Path prefix:   
All files starting with this prefix will be imported in lexicographic order

4 Data Preview | Path regex:   
Only files matching this regex will be included

5 Data Placement | Skip Glacier Objects:   
Skip processing objects stored in Amazon Glacier

Filter by Modified Time:

Start after path:   
Only paths lexicographically greater than this will be imported

Incremental?:   
When run repeatedly, attempt to only import new data since the last import. (Assumes lexicographic order)

[Cancel](#) [Back](#) [Next](#)

Parameters	Description
<b>Bucket</b>	<ul style="list-style-type: none"> <li>provide the S3 bucket name (Ex. <i>your_bucket_name</i>)</li> </ul>
<b>Path Prefix</b>	<ul style="list-style-type: none"> <li>specify a prefix for target keys. (Ex. <i>logs/data_</i>)</li> </ul>
<b>Path Regex</b>	<ul style="list-style-type: none"> <li>use regexp to match file paths. If a file path doesn't match the specified pattern, the file is skipped. For example, if you specify the pattern <i>.csv\$#</i> , then a file is skipped if its path doesn't match the pattern. Read more about <a href="#">regular expressions</a>.</li> </ul>
<b>Skip Glacier Objects</b>	<ul style="list-style-type: none"> <li>select to skip processing objects stored in the Amazon Glacier storage class. If objects are stored in Glacier storage class, but this option is not checked, an exception is thrown.</li> </ul>
<b>Filter by Modified Time</b>	<ul style="list-style-type: none"> <li>choose how to filter files for ingestion:</li> </ul>
If it is unchecked (default):	<ul style="list-style-type: none"> <li><b>Start after path:</b> inserts <i>last_path</i> parameter so that the first execution skips files before the path. (Ex. <i>logs/data_20170101.csv</i>)</li> <li><b>Incremental:</b> enables incremental loading. If incremental loading is enabled, config diff for the next execution includes the <i>last_path</i> parameter so that the next execution skips files before the path. Otherwise, <i>last_path</i> is not included.</li> </ul>
If it is checked:	<ul style="list-style-type: none"> <li><b>Modified after:</b> inserts <i>last_modified_time</i> parameters so that first execution skips files that were modified before that specified timestamp (Ex. <i>2019-06-03T10:30:19.806Z</i>)</li> <li><b>Incremental by Modified Time:</b> enables incremental loading by modified time. If incremental loading is enabled, config diff for the next execution includes the <i>last_modified_time</i> parameter so that the next execution skips files that were modified before that time. Otherwise, <i>last_modified_time</i> is not included.</li> </ul>

You can limit access to your S3 bucket/IAM user by using a list of static IPs. Contact [support@treasuredata.com](mailto:support@treasuredata.com) if you need static IPs.

There are instances where you might need to scan all the files in a directory (such as from the top-level directory "/"). In such instances, you must use the CLI to do the import.

### Example

Amazon CloudFront is a web service that speeds up the distribution of your static and dynamic web content. You can configure CloudFront to create log files that contain detailed information about every user request that CloudFront receives. If you enable logging, you can save CloudFront log files, shown as follows:

```
[your_bucket] - [logging] - [E231A697YXWD39.2017-04-23-15.a103fd5a.gz]
[your_bucket] - [logging] - [E231A697YXWD39.2017-04-23-15.b2aede4a.gz]
[your_bucket] - [logging] - [E231A697YXWD39.2017-04-23-16.594fa8e6.gz]
[your_bucket] - [logging] - [E231A697YXWD39.2017-04-23-16.d12f42f9.gz]
```

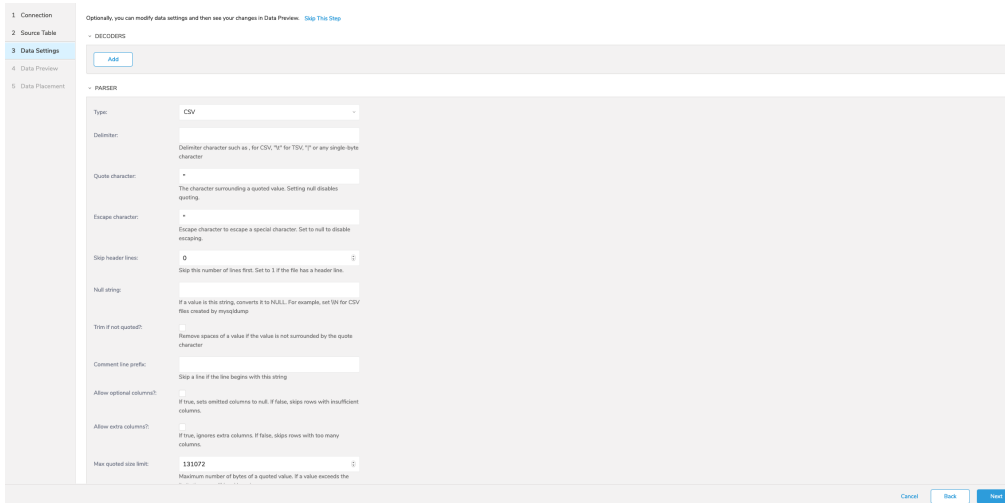
In this case, the Source Table settings are as shown:

- Bucket:** *your\_bucket*
- Path Prefix:** *logging/*
- Path Regex:** *.gz\$* (Not Required)
- Start after path:** *logging/E231A697YXWD39.2017-04-23-15.b2aede4a.gz* (Assuming that you want to import the log files from 2017-04-23-16.)
- Incremental:** *true* (if you want to schedule this job.)

BZip2 decoder plugin is supported as default. [Zip Decoder Function](#)

## Data Settings

- Select **Next**.  
The Data Settings page opens.
- Optionally, edit the data settings or skip this page of the dialog.



## Filters

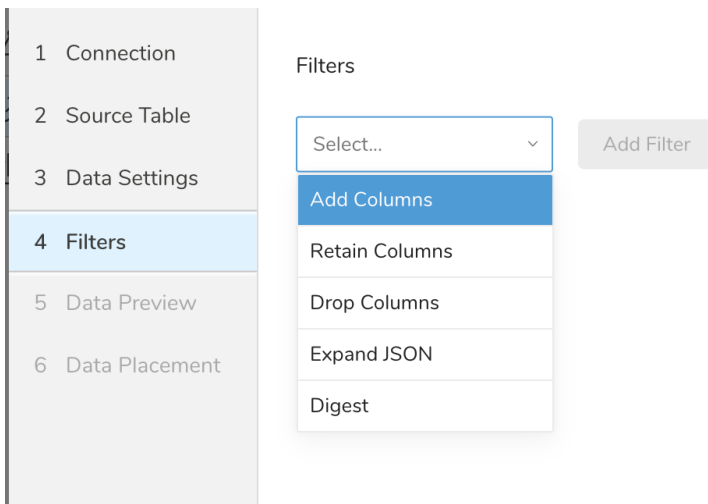
Import Integration Filters enable you to modify your imported data after you have completed [Editing Data Settings](#) for your import.

To apply import integration filters:

Select **Next** in Data Settings.

The Filters dialog opens.

Select the filter option you want to add.



Select **Add Filter**.

The parameter dialog for that filter opens.

Edit the parameters.

For information on each filter type, see one of the following:

- [Retaining Columns Filter](#)
- [Adding Columns Filter](#)
- [Dropping Columns Filter](#)
- [Expanding JSON Filter](#)
- [Digesting Filter](#)

Optionally, to add another filter of the same type, select **Add** within the specific column filter dialog.

Optionally, to add another filter of a different type, select the filter option from the list and repeat the same steps.

After you have added the filters you want, select **Next**.  
The Data Preview dialog opens.

## Data Preview

You can see a [preview](#) of your data before running the import by selecting Generate Preview.

Data shown in the data preview is approximated from your source. It is not the actual data that is imported.

1. Click **Next**.  
Data preview is optional and you can safely skip to the next page of the dialog if you want.
2. To preview your data, select **Generate Preview**. Optionally, click **Next**.
3. Verify that the data looks approximately like you expect it to.

The screenshot shows a 'Create Source' dialog box with a 'Data Preview' tab selected. The dialog title is 'Create Source Using onetrust\_demo'. The preview shows a subset of data from the source based on the data settings. The table has 8 columns: Ab\_id, Ab\_language, Ab\_identifier, last\_updated\_date, Ab\_link\_token, and two empty columns. The data is as follows:

	Ab_id	Ab_language	Ab_identifier	last_updated_date	Ab_link_token		
1	f7abf910-b5da-47c2-bbee-3714c86...	NULL	Quan3	2020-09-25 22:42:59...	NULL		
2	9022117f-cf3c-418c-b527-a8bd9a9...	NULL	Quan2	2020-08-05 03:48:19...	NULL		
3	a432b52f-3d93-483b-b65f-3c7530...	NULL	Quan4	2020-08-05 03:48:19...	NULL		
4	233ec0c2-70ab-4de4-ac48-a4a048f...	NULL	Quan5	2020-08-05 03:48:19...	NULL		
5	f78be70b-8b5d-404e-b663-b606a2...	NULL	Quan1	2020-08-05 03:48:19...	NULL		
6	db5d8f89-c264-4d82-a246-5939e5...	NULL	example@otrprivacy.com	2020-08-06 17:51:12...	NULL		
7	5ef9542c-315d-4b56-ad1c-c63ad0...	NULL	Michael.White@gmail.com	2020-09-09 20:01:45...	NULL		
8	3f1dfcb9-1904-4517-9087-0cc45f0...	NULL	Robert.Brown@gmail.com	2020-09-09 20:01:45...	NULL		
9	4a3a88dd-11a3-4c8b-a1d9-d7043f...	NULL	Mary.Anderson@mail.com	2020-09-09 20:01:46...	NULL		
10	4fd8983a-9e49-46dc-9519-1cf9dea...	NULL	Elizabeth.Scott@gmail.com	2020-09-09 20:01:47...	NULL		
11	33342e5d-4c95-4cfe-a622-4e91dc5...	NULL	David.Miller@aol.com	2020-09-09 20:01:47...	NULL		
12	f54bd07c-df75-4bf3-934a-dc19a96...	NULL	Robert.Anderson@att.com	2020-09-10 04:57:16...	NULL		
13	43bfe156-dfba-43b8-964d-1b2a4ae...	NULL	Elizabeth.Miller@google.com	2020-09-10 04:57:16...	NULL		

4. Select **Next**.

## Data Placement

For data placement, select the target database and table where you want your data placed and indicate how often the import should run.

1. Select **Next**. Under Storage you will create a new or select an existing database and create a new or select an existing table for where you want to place the imported data.

The screenshot shows the 'Data Placement' and 'Schedule' configuration panels in the Data Workbench. The 'Data Placement' panel includes a sidebar with steps 1-5, where '5 Data Placement' is selected. The main area shows settings for Database (chung\_default\_db), Table (sftp\_v2\_devproxy), Method (Append: Add records into existing table), Timestamp-based Partition Key (time), and Data Storage Timezone (UTC (default)). The 'Schedule' panel shows Repeat (Off) and Scheduling Timezone (Asia/Saigon).

2. Select a **Database** > **Select an existing** or **Create New Database**.
3. Optionally, type a database name.
4. Select a **Table**> **Select an existing** or **Create New Table**.
5. Optionally, type a table name.
6. Choose the method for importing the data.
  - **Append** (default)-Data import results are appended to the table. If the table does not exist, it will be created.
  - **Always Replace**-Replaces the entire content of an existing table with the result output of the query. If the table does not exist, a new table is created.
  - **Replace on New Data**-Only replace the entire content of an existing table with the result output when there is new data.
7. Select the **Timestamp-based Partition Key** column. If you want to set a different partition key seed than the default key, you can specify the long or timestamp column as the partitioning time. As a default time column, it uses upload\_time with the add\_time filter.
8. Select the **Timezone** for your data storage.
9. Under **Schedule**, you can choose when and how often you want to run this query.
  - Run once:
    - a. Select **Off**.
    - b. Select **Scheduling Timezone**.
    - c. Select **Create & Run Now**.
  - Repeat the query:
    - a. Select **On**.
    - b. Select the **Schedule**. The UI provides these four options: *@hourly*, *@daily* and *@monthly* or custom *cron*.
    - c. You can also select **Delay Transfer** and add a delay of execution time.
    - d. Select **Scheduling Timezone**.
    - e. Select **Create & Run Now**.

After your transfer has run, you can see the results of your transfer in **Data Workbench** > **Databases**.

## Validating Your Data Connector Jobs

### How do I troubleshoot data import problems?

Review the job log. Warning and errors provide information about the success of your import. For example, you can [identify the source file names associated with import errors](#).

To find out more about a specific job, you can select that job and see details. Depending on the type of job, you can see some or all of the following: results, query, output logs, engine logs, details, and destination.

1. Open the TD Console.
2. Navigate to **Jobs**. You can review the number of jobs which is listed in the upper right of the page.

Jobs		Job Activities	
Status	Job	Started	
QUEUED	10349483 - Data Import { "embulk_config": { "in": { "client_secre...		
SUCCESS 9 secs	10349473 - Presto DROP TABLE IF EXISTS "cdp_tmp_activit...	5:05 pm	
SUCCESS 9 secs	10349472 - Presto DROP TABLE IF EXISTS "cdp_tmp_partiti...	5:05 pm	
SUCCESS 2 secs	10349470 - Presto DROP TABLE IF EXISTS "cdp_tmp_scored_...	5:05 pm	
SUCCESS 14 secs	10349457 - Result Export RESULT EXPORT FROM JOB 10349456	5:04 pm	

- Optionally, use filters to reduce the listing of jobs to locate what you are interested in. Including filtering by job owner, date, and database name.
- Select a job to open it and view results, query definition, logs, and other details.

Jobs **Job Activities / 10349473** SUCCESS COPY TO CLIPBOARD DOWNLOAD

Results Query Output Logs Engine Logs Details

- Each tab has different information about the job.

**Job Activities / 616502369** SUCCESS

Results Query Output Logs Engine Logs Details

**Job Activities / 18712917** SUCCESS

Query Output Logs Details Destination

Results	<ul style="list-style-type: none"> <li>View the imported data from the job.</li> <li>From here you can copy the results to the clipboard or download them as a CSV file.</li> </ul>
Query	<ul style="list-style-type: none"> <li>View the query syntax of the job</li> <li>Launch a query editor</li> <li>Copy queries and use to create new queries or workflows</li> <li>Refine queries to improve efficiency</li> </ul>



Output and Engine Logs	<ul style="list-style-type: none"> <li>• Log information can be reviewed for run times, query result numbers, and error codes</li> <li>• Log information can be copied to the clipboard</li> </ul>
Details	<p>View further details:</p> <ul style="list-style-type: none"> <li>• query name</li> <li>• type</li> <li>• job id</li> <li>• status</li> <li>• duration</li> <li>• scheduled and actual times</li> <li>• result count and size</li> <li>• runner,</li> <li>• database queried</li> <li>• priority</li> </ul>
Destination	<p>Here you can view details of an export integration configuration:</p> <ul style="list-style-type: none"> <li>• integration</li> <li>• type</li> <li>• settings</li> </ul>

## What can I do if the data connector for S3 job is running for a long time?

Check the count of S3 files that your connector job is ingesting. If there are over 10,000 files, the performance degrades. To mitigate this issue, you can:

- Narrow path\_prefix option and reduce the count of S3 files.
- Set 268,435,456 (256MB) to min\_task\_size option.

## Sample Workflow

There is a sample workflow file for S3 import integration. You can define the import settings using yml file, and run it using `td\_load>` workflow operator. Variable definitions that cannot be used with the Source function of the TD console alone are possible with yml file-based execution.

You can refer the sample code from [https://github.com/treasure-data/treasure-boxes/tree/master/td\\_load/s3](https://github.com/treasure-data/treasure-boxes/tree/master/td_load/s3).

```

timezone: UTC

schedule:
  daily>: 02:00:00

sla:
  time: 08:00
  +notice:
    mail>: {data: Treasure Workflow Notification}
    subject: This workflow is taking long time to finish
    to: [me@example.com]

_export:
  td:
    dest_db: dest_db_ganesh
    dest_table: dest_table_ganesh

+prepare_table:
  td_ddl>:
    create_databases: ["${td.dest_db}"]
    create_tables: ["${td.dest_table}"]
    database: ${td.dest_db}

+load:
  td_load>: config/daily_load.yml
  database: ${td.dest_db}
  table: ${td.dest_table}

```