

# Package ‘backblazer’

August 29, 2016

**Type** Package

**Title** Bindings to the Backblaze B2 API

**Version** 0.1.0

**Date** 2016-01-29

**Author** Authors@R: person(“Phill”, “Clarke”, email = “phill@starkingdom.co.uk”,  
role = c(“aut”, “cre”))

**Maintainer** Phill Clarke <phill@starkingdom.co.uk>

**Description** Provides convenience functions for the Backblaze B2 cloud storage  
API (see <https://www.backblaze.com/b2/docs/>). All B2 API calls are mapped  
to equivalent R functions. Files can be easily uploaded, downloaded and  
deleted from B2, all from within R programs.

**Depends** R (>= 3.2.3)

**License** MIT + file LICENSE

**URL** <https://github.com/phillc73/backblazer>

**BugReports** <http://www.github.com/phillc73/backblazer/issues>

**LazyData** TRUE

**Imports** httr (>= 1.0.0), jsonlite (>= 0.9.19), openssl (>= 0.9.1),  
tools (>= 3.2.3)

**Suggests** knitr (>= 1.12), rmarkdown

**VignetteBuilder** knitr

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2016-01-31 16:36:49

## R topics documented:

b2AuthorizeAccount . . . . .	2
b2CreateBucket . . . . .	3

b2DeleteBucket . . . . .	4
b2DeleteFileVersion . . . . .	5
b2DownloadFileById . . . . .	6
b2DownloadFileByName . . . . .	7
b2GetFileInfo . . . . .	8
b2GetUploadUrl . . . . .	9
b2HideFile . . . . .	10
b2ListBuckets . . . . .	11
b2ListFileNames . . . . .	11
b2ListFileVersions . . . . .	12
b2UpdateBucket . . . . .	14
b2UploadFile . . . . .	15
<b>Index</b>	<b>17</b>

---

b2AuthorizeAccount	<i>Authorise Backblaze B2 Account.</i>
--------------------	--

---

## Description

b2AuthorizeAccount authorises an account on Backblaze's B2 cloud storage product.

## Usage

```
b2AuthorizeAccount(url, accountId, authorizationKey)
```

## Arguments

url	Specific API endpoint address for this function. See examples.
accountId	Account identification code for the relevant Backblaze B2 account. This may be obtained by clicking <i>Show Account ID and Application Key</i> hypertext from the B2 My Account area, after logging in with a web browser.
authorizationKey	Account authorisation key for the relevant Backblaze B2 account. This may be obtained by clicking <i>Show Account ID and Application Key</i> hypertext from the B2 My Account area, after logging in with a web browser.

## Details

This authorisation function **must be executed first**, before any other functions in this package. Failure to execute b2AuthorizeAccount renders everything else pointless. You will require a valid Backblaze B2 accountId and authorizationKey. Create a Backblaze B2 account and obtain your access credentials here:

<https://www.backblaze.com/b2/cloud-storage.html>

Further documentation regarding the Backblaze B2 Cloud Storage API is available here:

<https://www.backblaze.com/b2/docs/>

API account authorization url, accountId, authorizationKey are all mandatory and must be user defined.

Every time b2AuthorizeAccount is executed, a new login to Backblaze B2 occurs. Don't login more than is necessary.

### Value

If successful, an authorisation token will be returned and stored in an Rds file called *accountAuthorization.Rds* in the current working directory. The data in this Rds file will be used in all other functions in this package. Specific B2 documentation regarding this API call can be found here:

[https://www.backblaze.com/b2/docs/b2\\_authorize\\_account.html](https://www.backblaze.com/b2/docs/b2_authorize_account.html)

### Note

Consider programmatically deleting *accountAuthorization.Rds* on exit.

### Examples

```
## Not run:
b2AuthorizeAccount(url = "https://api.backblaze.com/b2api/v1/b2_authorize_account",
  accountId = "YourAccountId",
  authorizationKey = "YourAuthorisationKey")

## End(Not run)
```

---

b2CreateBucket	<i>Create B2 Bucket.</i>
----------------	--------------------------

---

### Description

b2CreateBucket creates a new bucket in the user's account on the Backblaze B2 cloud storage product.

### Usage

```
b2CreateBucket(bucketName, bucketType)
```

### Arguments

bucketName	Bucket names must be globally unique. No two users may have buckets named the same. Bucket names may not start with <i>b2</i> . Bucket names must be a minimum of 6 and maximum of 50 characters long. Bucket names may consist only of letters, numbers and hyphens. Special characters are invalid.
bucketType	Supported bucket types are <i>allPublic</i> and <i>allPrivate</i> .

### Details

This function creates a new bucket within the user's account on the Backblaze B2 cloud storage product. Backblaze B2 does not support tree based folder structures as such, meaning all uploaded data is stored in a flat file structure. However, buckets may be created at the top level to aid in content organisation. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_create\\_bucket.html](https://www.backblaze.com/b2/docs/b2_create_bucket.html)

bucketName and bucketType are mandatory and must be user defined.

### Value

If successful a list containing the accountId, bucketName and bucketType will all be echoed back to the user. Also included in the return list will be a unique bucketId.

### Examples

```
## Not run:
b2CreateBucket(bucketName = "this-is-a-uniquely-named-bucket",
bucketType = "allPublic")

## End(Not run)
```

---

b2DeleteBucket	<i>Delete B2 Bucket.</i>
----------------	--------------------------

---

### Description

b2DeleteBucket deletes an existing bucket in the user's account on the Backblaze B2 cloud storage product.

### Usage

```
b2DeleteBucket(bucketId)
```

### Arguments

bucketId	The unique identifier of the bucket to be deleted. A list of all the user's bucket IDs may be found using the <code>b2_list_buckets</code> function in this package.
----------	--

### Details

This function deletes an existing bucket within the user's account on the Backblaze B2 cloud storage product. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_delete\\_bucket.html](https://www.backblaze.com/b2/docs/b2_delete_bucket.html)

bucketId is mandatory and must be user defined.

**Value**

If successful a list containing the accountId, bucketId, bucketName and bucketType of the deleted bucket will all be echoed back to the user.

**Examples**

```
## Not run:  
b2DeleteBucket(bucketId = "aUniqueBucketId")  
  
## End(Not run)
```

---

b2DeleteFileVersion     *Delete B2 File Version.*

---

**Description**

b2DeleteFileVersion deletes a version of a file in the user's account on the Backblaze B2 cloud storage product.

**Usage**

```
b2DeleteFileVersion(fileName, fileId)
```

**Arguments**

fileName	The name of the file to be deleted. File names may be obtained through the b2ListFileNames function in this package.
fileId	The unique identifier of the file to be deleted. File IDs may be obtained through the b2UploadFile, b2ListFileNames, or b2ListFileVersions functions in this package.

**Details**

This function deletes a version of a file within the user's account on the Backblaze B2 cloud storage product. Files of the same name may have multiple versions stored on B2. If the deleted file version is the latest, and there are older versions of the same file, the most recent of these will become the latest version. The most recent file version is always the version downloaded if requested by name. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_delete\\_file\\_version.html](https://www.backblaze.com/b2/docs/b2_delete_file_version.html)

fileName and fileId are mandatory and must be user defined.

**Value**

If successful a list containing the fileId and fileName will be echoed back to the user.

**Examples**

```
## Not run:
b2DeleteFileVersion(fileName = "nameOfTheFileToDelete",
  fileId = "Unique_identifier_of_the_file_to_delete")

## End(Not run)
```

---

b2DownloadFileById      *Download B2 File by ID.*

---

**Description**

b2DownloadFileById downloads a file from the user's account on the Backblaze B2 cloud storage product.

**Usage**

```
b2DownloadFileById(fileId, overwrite = FALSE)
```

**Arguments**

fileId	The unique identifier of the file to be downloaded. File IDs may be obtained through the b2ListFiles, b2ListFileVersions and b2UploadFile functions in this package.
overwrite	Binary TRUE or FALSE decision to overwrite any files in the current working directory, whose names match the downloaded file name.

**Details**

This function downloads a file from the user's account on the Backblaze B2 cloud storage product using the file's unique ID only. Files of the same name may have multiple versions stored on B2. Therefore, every file version will have a unique ID, which can be used for downloading that specific version. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_download\\_file\\_by\\_id.html](https://www.backblaze.com/b2/docs/b2_download_file_by_id.html)

fileId is mandatory and must be user defined. overwrite is optionally user defined and defaults to FALSE.

**Value**

If successful the response headers include the Content-Type that was specified when the file was uploaded. They also include the X-Bz-FileName and X-Bz-Content-Sha1 headers. The X-Bz-FileName uses percent-encoding, as if it were a URL parameter. If successful, the file will be downloaded to the current working directory.

## Examples

```
## Not run:  
b2DownloadFileById(fileId = "Unique_identifier_of_the_file_to_download",  
  overwrite = TRUE)  
  
## End(Not run)
```

---

b2DownloadFileByName *Download B2 File by Name.*

---

## Description

b2DownloadFileByName downloads a file from the user's account on the Backblaze B2 cloud storage product.

## Usage

```
b2DownloadFileByName(bucketName, fileName, overwrite = FALSE)
```

## Arguments

bucketName	The name of the bucket containing the requested file. Bucket names may be obtained through the b2ListBuckets function in this package.
fileName	The name of the file to be downloaded. File names may be obtained through the b2ListFileNames function in this package.
overwrite	Binary TRUE or FALSE decision to overwrite any files in the current working directory, whose names match the downloaded file name.

## Details

This function downloads a file from the user's account on the Backblaze B2 cloud storage product using the file's name only. Files of the same name may have multiple versions stored on B2. Therefore, only the most recent version matching the specified filename will be downloaded. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_download\\_file\\_by\\_name.html](https://www.backblaze.com/b2/docs/b2_download_file_by_name.html)

fileName and bucketName are mandatory and must be user defined. overwrite is optionally user defined and defaults to FALSE.

## Value

If successful the response headers include the Content-Type that was specified when the file was uploaded. They also include the X-Bz-FileName and X-Bz-Content-Sha1 headers. The X-Bz-FileName uses percent-encoding, as if it were a URL parameter. If successful, the file will be downloaded to the current working directory.

## Examples

```
## Not run:  
b2DownloadFileByName(bucketName = "this-is-a-uniquely-named-bucket",  
  fileName = "yourFileName.txt", overwrite = TRUE)  
  
## End(Not run)
```

---

b2GetFileInfo

*B2 Get File Info.*

---

## Description

b2GetFileInfo returns information about a single file from the user's account on the Backblaze B2 cloud storage product.

## Usage

```
b2GetFileInfo(fileId)
```

## Arguments

fileId	The unique identifier of the file whose information is required. File IDs may be obtained through the b2ListFiles, b2ListFileVersions and b2UploadFile functions in this package.
--------	---

## Details

This function returns information about a single file from the user's account on the Backblaze B2 cloud storage product. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_get\\_file\\_info.html](https://www.backblaze.com/b2/docs/b2_get_file_info.html)

fileId is mandatory and must be user defined.

## Value

If successful a list will be returned containing fileId, fileName, accountId, contentSha1, bucketId, contentLength, contentType and fileInfo.

## Examples

```
## Not run:  
b2GetFileInfo(fileId = "Unique_identifier_of_the_file")  
  
## End(Not run)
```



---

b2GetUploadUrl	<i>Get B2 Upload URL.</i>
----------------	---------------------------

---

### Description

b2GetUploadUrl returns the URL required in order to upload files to the user's account on the Backblaze B2 cloud storage product.

### Usage

```
b2GetUploadUrl(bucketId)
```

### Arguments

bucketId	The unique identifier of the bucket where files are to be uploaded. Bucket IDs may be obtained through the b2ListBucketsfunction in this package.
----------	---

### Details

This function returns the URL required in order to upload files the user's account on the Backblaze B2 cloud storage product. An uploadUrl and upload authorizationToken are also returned. These are valid for 24 hours or until the endpoint rejects an upload, Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_get\\_upload\\_url.html](https://www.backblaze.com/b2/docs/b2_get_upload_url.html)

bucketId is mandatory and must be user defined.

### Value

If successful a list will be returned containing bucketId, uploadURL and authorizationToken.

### Examples

```
## Not run:
uploadUrlReturn <- b2GetUploadUrl(bucketId = "aUniqueBucketId")
uploadUrl <- uploadUrlReturn$uploadUrl
authToken <- uploadUrlReturn$authorizationToken

## End(Not run)
```

---

b2HideFile	<i>Hide B2 File.</i>
------------	----------------------

---

### Description

b2HideFile hides a file in the user's account on the Backblaze B2 cloud storage product.

### Usage

```
b2HideFile(bucketId, fileName)
```

### Arguments

bucketId	The unique identifier of the bucket containing the file to be hidden. Bucket IDs may be obtained through the b2ListBuckets function in this package.
fileName	The name of the file to be hidden. File names may be obtained through the b2ListFileNames function in this package.

### Details

This function hides a file in the user's account on the Backblaze B2 cloud storage product, so that it cannot be downloaded by name. In order to further understand the concept of B2 file versions, see the specific Backblaze documentation. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_hide\\_file.html](https://www.backblaze.com/b2/docs/b2_hide_file.html) [https://www.backblaze.com/b2/docs/file\\_versions.html](https://www.backblaze.com/b2/docs/file_versions.html)

bucketId fileName are mandatory and must be user defined.

### Value

If successful a list will be returned containing fileId, fileName, action, size and uploadTimestamp.

### Examples

```
## Not run:  
b2HideFile(bucketId = "aUniqueBucketId", fileName = "yourFileName.txt")  
  
## End(Not run)
```

---

b2ListBuckets	<i>List B2 Buckets.</i>
---------------	-------------------------

---

**Description**

b2ListBuckets lists buckets associated with an account, in alphabetical order by bucket ID.

**Usage**

```
b2ListBuckets()
```

**Details**

This function lists buckets associated with a user's account on the Backblaze B2 cloud storage product, in alphabetical order by bucket ID. Further details regarding this API call are available [here](https://www.backblaze.com/b2/docs/b2_list_buckets.html):

[https://www.backblaze.com/b2/docs/b2\\_list\\_buckets.html](https://www.backblaze.com/b2/docs/b2_list_buckets.html)

b2ListBuckets does not support any function options. However, the function must first be authorised through the b2AuthorizeAccount function.

**Value**

If successful a data frame will be returned containing accountId, bucketId, bucketName and bucketType for all existing buckets in the specific user's account.

**Examples**

```
## Not run:  
b2ListBuckets()  
  
## End(Not run)
```

---

b2ListFileNames	<i>List B2 File Names.</i>
-----------------	----------------------------

---

**Description**

b2ListFileNames lists the names of all files in a bucket, starting at a given name.

**Usage**

```
b2ListFileNames(bucketId, startFileName = "", maxFileCount = 100)
```

**Arguments**

bucketId	The unique identifier of the bucket containing the files to be listed. Bucket IDs may be obtained through the b2ListBucketsfunction in this package.
startFileName	The name of the file from which the list will start. This is an optional parameter. Not defining this parameter will result in the list starting from the newest file first. File names may be obtained through the b2ListFileNames function in this package.
maxFileCount	An integer defining the maximum number of file names to return. This is an optional parameter and defaults to 100. The maximum acceptable value is 1000.

**Details**

This function lists the names of all files in a bucket, starting at a given name, within a user's account on the Backblaze B2 cloud storage product. There may be many file versions for the same name, but this function will return each name only once. If you want all of the versions, use b2ListFileVersions instead. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_list\\_file\\_names.html](https://www.backblaze.com/b2/docs/b2_list_file_names.html)

bucketId is mandatory and must be user defined. startFileName and maxFileCount are optional and may be defined by the user if so desired.

**Value**

If successful a list will be returned containing files and nextFileName, for files within the specified bucket. If greater than the maximum number of specified files in maxFileCount exists, further file names may be obtained beginning with nextFileName, as the startFileName parameter value. If successful a data frame will be returned, nested within files, containing fileId, fileName, action, size and uploadTimestamp for all files within the specified bucket.

**Examples**

```
## Not run:
b2ListFileNames(bucketId = "aUniqueBucketId",
startFileName = "yourFileName.png",
maxFileCount = 500)

## End(Not run)
```

---

b2ListFileVersions      *List B2 File Versions.*

---

**Description**

b2ListFileVersions lists all versions of all of the files contained in one bucket, within a user's account on the Backblaze B2 cloud storage product.

## Usage

```
b2ListFileVersions(bucketId, startFileName = "", startFileId = "",
  maxFileCount = 100)
```

## Arguments

bucketId	The unique identifier of the bucket containing the files to be listed. Bucket IDs may be obtained through the <code>b2ListBuckets</code> function in this package.
startFileName	The name of the file from which the list will start. If there are no files with this name, the first version of the file with the first name after the given name will be the first in the list. This is an optional parameter. Not defining this parameter will result in the list starting from the newest file first. File names may be obtained through the <code>b2ListFileNames</code> function in this package.
startFileId	The first file ID to return. This is an optional parameter. <code>startFileName</code> must also be provided if <code>startFileId</code> is specified.
maxFileCount	An integer defining the maximum number of file names to return. This is an optional parameter and defaults to 100. The maximum acceptable value is 1000.

## Details

This function lists all versions of all of the files contained in one bucket, within a user's account on the Backblaze B2 cloud storage product. Files will be listed in alphabetical order by file name, and by reverse of date/time uploaded for versions of files with the same name. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_list\\_file\\_versions.html](https://www.backblaze.com/b2/docs/b2_list_file_versions.html)

`bucketId` is mandatory and must be user defined. `startFileName`, `startFileId` and `maxFileCount` are optional and may be defined by the user if so desired.

## Value

If successful a list will be returned containing `files`, `nextFileName` and `nextFileId` for files within the specified bucket. If greater than the maximum number of specified files in `maxFileCount` exists, further file versions may be obtained beginning with `nextFileName`, as the `startFileName` parameter value. This may be done either alone or in combination with `nextFileId`, as `startFileId`. If successful a data frame will be returned, nested within `files`, containing `fileId`, `fileName`, `action`, `size` and `uploadTimestamp` for all file versions within the specified bucket.

## Examples

```
## Not run:
b2ListFileVersions(bucketId = "aUniqueBucketId",
  startFileName = "yourFileName.png",
  startFileId = "yourFileId"
  maxFileCount = 500)

## End(Not run)
```

---

b2UpdateBucket	<i>Update B2 Bucket.</i>
----------------	--------------------------

---

### Description

b2UpdateBucket modifies the bucket type of an existing bucket in the user's account.

### Usage

```
b2UpdateBucket(bucketId, bucketType)
```

### Arguments

bucketId	The unique identifier of the bucket to be updated. A list of all the user's bucket IDs may be found using the <code>b2_list_buckets</code> function in this package.
bucketType	Supported bucket types are <i>allPublic</i> and <i>allPrivate</i> .

### Details

This function modifies the bucket type of an existing bucket, within the user's account on the Backblaze B2 cloud storage product. This function can be used to allow everyone to download the contents of the bucket without providing any authorisation, or to prevent anyone from downloading the contents of the bucket without providing a bucket authToken. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_update\\_bucket.html](https://www.backblaze.com/b2/docs/b2_update_bucket.html)

bucketId and bucketType are mandatory and must be user defined.

### Value

If successful a list containing the accountId, bucketId, bucketName and bucketType will all be echoed back to the user.

### Examples

```
## Not run:  
# Make a bucket Private  
b2UpdateBucket(bucketId = "aUniqueBucketId", bucketType = "allPrivate")  
# Make a bucket Public  
b2UpdateBucket(bucketId = "aUniqueBucketId", bucketType = "allPublic")  
  
## End(Not run)
```

---

b2UploadFile	<i>Upload File to B2.</i>
--------------	---------------------------

---

**Description**

b2Uploadfile Uploads one file to the user's account B2.

**Usage**

```
b2UploadFile(authToken, uploadUrl, fileName)
```

**Arguments**

authToken	A unique authorisation token permitting the upload of files to B2. An authorisation token is obtained from b2UploadUrl.
uploadUrl	A unique URL permitting the upload of files to B2. An upload URL is obtained from b2UploadUrl.
fileName	The name of the file to be uploaded. Files must be located in the current working directory. A complete list of supported file types may be found in the b2FileTypes.rds file located in this package's data directory, or at the following location: <a href="https://www.backblaze.com/b2/docs/content-types.html">https://www.backblaze.com/b2/docs/content-types.html</a>

**Details**

This function uploads one file to the user's account on the Backblaze B2 cloud storage product, returning a unique file ID. Files cannot be uploaded to B2 unless GetUploadUrl has been executed first, specifying the bucket in which to upload the file. Further details regarding this API call are available here:

[https://www.backblaze.com/b2/docs/b2\\_upload\\_file.html](https://www.backblaze.com/b2/docs/b2_upload_file.html)

authToken, uploadUrl and fileName are mandatory and must be user defined.

**Value**

If successful a list containing the fileId, fileName, accountId, bucketId, contentLength, contentSha1, contentType and fileInfo will all be echoed back to the user.

**Examples**

```
## Not run:
# Make a bucket Private
# Get Upload URL
uploadUrlReturn <- b2GetUploadUrl(bucketId = "aUniqueBucketId")
uploadUrl <- uploadUrlReturn$uploadUrl
authToken <- uploadUrlReturn$authorizationToken
# Upload file
b2UploadFile(authToken, uploadUrl, fileName = "yourFileName.png")
```

## End(Not run)



# Index

b2AuthorizeAccount, [2](#)  
b2CreateBucket, [3](#)  
b2DeleteBucket, [4](#)  
b2DeleteFileVersion, [5](#)  
b2DownloadFileById, [6](#)  
b2DownloadFileByName, [7](#)  
b2GetFileInfo, [8](#)  
b2GetUploadUrl, [9](#)  
b2HideFile, [10](#)  
b2ListBuckets, [11](#)  
b2ListFileNames, [11](#)  
b2ListFileVersions, [12](#)  
b2UpdateBucket, [14](#)  
b2UploadFile, [15](#)