

Authentication

Login to Keystone in OpenStack Swift.

URL structure

<http://cloudspaces.urv.cat:5000/v2.0/tokens>

Method

GET

Headers

- Content-Type: application/json

Request Body

```
{
  "auth":
  {
    "passwordCredentials":
    {
      "username": "userX",
      "password": "userX"
    },
    "tenantName": "userX"
  }
}
```

Response Body

```
{
  "access":{
    "token":{
      "issued_at":"2013-10-03T10:10:18.705131",
      "expires":"2013-10-04T10:10:18Z",
      "id":"MIIGewYeMN-M=",
      "tenant":{
        "description":"Description userX",
        "enabled":true,
        "id":"0cd10527bafb4d778ba5a3a5bb419828",
        "name":"userX"
      }
    },
    "serviceCatalog":[
      {
        "endpoints":[
          {
```

```

        "adminURL":"https://cloudspaces.urv.cat:8080/v1",
        "region":"RegionOne",
        "internalURL":"https://cloudspaces.urv.cat:8080/v1/AUTH_0cd10527bafb4d778ba5a3a5bb419828",
        "id":"61807a44e9e141e998b67814bb1103c7",
        "publicURL":"https://cloudspaces.urv.cat:8080/v1/AUTH_0cd10527bafb4d778ba5a3a5bb419828"
    },
    ],
    "endpoints_links":[
    ],
    "type":"object-store",
    "name":"swift"
},
{
    "endpoints":[
        {
            "adminURL":"http://cloudspaces.urv.cat:35357/v2.0",
            "region":"RegionOne",
            "internalURL":"http://cloudspaces.urv.cat:5000/v2.0",
            "id":"1c77111ccbcc429a960d2bed06ea2942",
            "publicURL":"http://cloudspaces.urv.cat:5000/v2.0"
        }
    ],
    "endpoints_links":[
    ],
    "type":"identity",
    "name":"keystone"
}
],
"user":{
    "username":"userX",
    "roles_links":[
    ],
    "id":"85f0e8e483cb43e1b6a97d3cc27a58cb",
    "roles":[
        {
            "name":"_member_"
        },
        {
            "name":"rusers"
        }
    ],
    "name":"userX"
},
"metadata":{
    "is_admin":0,
    "roles":[
        "9fe2ff9ee4384b1894a90878d3e92bab",
        "a0eeaa6045c54d0d8e3dcf35c8aaf1ab"
    ]
}
}
}

```

Important values are:

- **access.token.id**: The access token that will be attached in every request to obtain access to API resources.
- **access.serviceCatalog**: It is a list that contains all services available. The one interesting for the API is Swift. Therefore, the only item on the list we need is the one that contains `swift` in the `name` parameter.
- **access.serviceCatalog[X].endpoints[0].publicURL**: The base URL where the API is located.

Error handling

Errors are returned using standard HTTP error code syntax. Any additional info is included in the body of the return call, JSON-formatted. Error codes not listed here are in the REST API methods listed below.

Standard API errors

Code	Description
400	Bad input parameter. Error message should indicate which one and why.
401	Authorization required. The presented credentials, if any, were not sufficient to access the folder resource. Returned if an application attempts to use an access token after it has expired.
403	Forbidden. The requester does not have permission to access the specified resource.
404	File or folder not found at the specified path.
405	Request method not expected (generally should be GET or POST).
5xx	Server error

File

Create a file

An application can create a file by issuing an HTTP POST request. In addition, the application needs to provide as input, JSON that identifies the name and the folder that contains the file to be created. The location of the newly created file is returned in the Location response header.

Note that an application can create a file inside a folder or directly in the root omitting the parent field.

Request

URL structure

The URL that represents the file data resource. The URL is `https://domain.ext/stacksync/file`.

Method

POST

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.
Content-Length	The length of the request body.
Content-Type	The content type and character encoding of the response. The content type must be <code>application/xml</code> , and the character encoding must be UTF-8.

Request Body

JSON input that contains a dictionary with the following keys:

ELEMENT	DESCRIPTION	REQUIRED
name	The user-visible name of the file to be created.	Yes

parent	ID of the folder where the file is going to be created. If no ID is passed, it will use the top-level folder. This parameter should not point to a file.	No
--------	---	----

HTTP Request Example

```
POST https://domain.ext/stacksync/file
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
Content-Length: 294
Content-Type: application/json
```

```
{
  "name": "Winter2012.jpg",
  "parent": 12386548974
}
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.
Location	The location of the newly created file.

Response Body

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
parent	ID of the folder's parent.

Response example

HTTP/1.1 201 Created

Content-Type: application/json; charset=UTF-8

Content-Length: 248

Location: <https://domain.ext/stacksync/file/32565632156>

```
{
  "name": "Client1.pdf",
  "path": "/documents/clients/Client1.pdf",
  "id": 32565632156,
  "parent": -348534824681,
  "user": "Adrian"
}
```

Upload file data

An application can upload data to a file by issuing an HTTP PUT request to the file data resource that represents the data for the file.

Uploading data to a file creates a new file version in the StackSync datastore and associates the uploaded data with the newly created file version.

This approach does not allow the application to resume the upload if the upload does not successfully complete, that is, if all the data is not completely uploaded.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, followed by the file ID and ends with `/data`, for example, `https://domain.ext/stacksync/file/2148742318/data`.

Method

PUT

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.
Content-Length	The content length of the input in bytes. Chunked transfer encoding can be specified if the content length is not known at the time the request is initiated.

HTTP Request Example

```
PUT https://domain.ext/stacksync/file/2148742318/data
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
Content-Length: 1478
```


<File content.... >

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content (if any).
Content-Type	The content type and character encoding of the response.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

Download file data

To retrieve file data, an application submits an HTTP GET request to the file data resource that represents the data for the file.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with `/data`, for example, `https://domain.ext/stacksync/file/2148742318/data`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/file/2148742318/data
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content (if any).

Content-Type	The content type and character encoding of the response.
--------------	--

Response Body

The response body contains the retrieved file data.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248

<File content.... >
```

Delete a file

An application can delete a file by issuing an HTTP DELETE request to the URL of the file resource. It's a good idea to precede DELETE requests like this with a caution note in your application's user interface.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, for example, `https://domain.ext/stacksync/file/2148742318`.

Method

DELETE

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
DELETE https://domain.ext/stacksync/file/2148742318
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
-------	-------------

Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.

Response Body

The response body contains the associated metadata of the deleted file:

ELEMENT	DESCRIPTION
name	The user-visible name of the file.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
checksum	The file’s checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
{
  "name": "Client1.pdf",
```

```
"path":"/documents/clients/Client1.pdf",  
"id":32565632156,  
"size":775412,  
"mimetype":"application/pdf",  
"status":"DELETED",  
"version":3,  
"parent":-348534824681,  
"user":"Adrian",  
"client_modified":"2013-03-08 10:36:41.997",  
"server_modified":"2013-03-08 10:36:41.997"  
}
```

Get file metadata

To retrieve information about a file, an application submits an HTTP GET request to the file resource that represents the file.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with the file ID, for example, `https://domain.ext/stacksync/file/2148742318`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/file/2148742318
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.

Content-Type	The content type and character encoding of the response.
--------------	--

Response Body

The response body contains the associated metadata of the file in JSON:

ELEMENT	DESCRIPTION
name	The user-visible name of the fil.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
checksum	The file’s checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
{
  "name": "Client1.pdf",
  "path": "/documents/clients/Client1.pdf",
  "id": 32565632156,
```



```
"size":775412,  
"mimetype":"application/pdf",  
"status":"DELETED",  
"version":3,  
"parent":-348534824681,  
"user":"Adrian",  
"client_modified":"2013-03-08 10:36:41.997",  
"server_modified":"2013-03-08 10:36:41.997"  
}
```

Update file metadata (move and rename)

An application can update various attributes of a file by issuing an HTTP PUT request to the URL that represents the file resource. In addition, the app needs to provide as input, JSON that identifies the new attribute values for the file. Upon receiving the PUT request, the StackSync service examines the input and updates any of the attributes that have been modified.

Here are the file attributes that can be updated:

- name
- parent

No other file attributes can be modified using a PUT request.

Note: An application can move a file to a different parent folder by changing the value of the parent element.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with the file ID, for example, `https://domain.ext/stacksync/file/2148742318`.

Method

PUT

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.
Content-Length	The length of the request body.
Content-Type	The content type and character encoding of the response. The content type must be <code>application/xml</code> , and the character encoding must be UTF-8.

Request Body

JSON input that contains a dictionary with the following keys:

ELEMENT	DESCRIPTION	REQUIRED
name	The user-visible name of the file to be created.	No
parent	ID of the folder where the file or folder is going to be uploaded. If no ID is passed, it will use the top-level folder. This parameter should not point to a file.	No

HTTP Request Example

```
POST https://domain.ext/stacksync/file/32565632156
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
Content-Length: 294
Content-Type: application/json
```

```
{
  "name": "Winter2012_renamed.jpg",
  "parent": 12386548974
}
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.
Location	The location of the newly created file.

Response Body

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are "NEW", "CHANGED", "DELETED". Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder's parent.
checksum	The file's checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
Location: https://domain.ext/stacksync/file/
```

```
{
  "name": "Winter2012_renamed.jpg",
  "path": "/documents/clients/Client1.pdf",
  "id": 32565632156,
  "size": 775412,
  "mimetype": "application/pdf",
```

```
"status":"CHANGED",  
"version":2,  
"parent":12386548974,  
"user":"Adrian",  
"client_modified":"2013-03-08 10:36:41.997",  
"server_modified":"2013-03-08 10:36:41.997"  
}
```

Versions

Get file versions

To retrieve information about a file version, an application submits an HTTP GET request to the file version resource.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with `/version`, for example, `https://domain.ext/stacksync/file/2148742318/version`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/file/2148742318/version
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
-------	-------------

Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.

Response Body

Contains a JSON list with a dictionary in each position with the following information:

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are “NEW”, “CHANGED”, “RENAMED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
checksum	The file’s checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
[
  {
```

```
    "name": "Winter2012.jpg",
    "path": "/documents/clients/Winter2012.jpg",
    "id": 32565632156,
    "size": 775412,
    "mimetype": "image/jpeg",
    "status": "NEW",
    "version": 1,
    "parent": 12386548974,
    "user": "Adrian",
    "client_modified": "2013-03-08 10:36:41.997",
    "server_modified": "2013-03-08 10:36:41.997"
  },
  {
    "name": "Winter2012.jpg",
    "path": "/documents/clients/Winter2012.jpg",
    "id": 32565632156,
    "size": 7482,
    "mimetype": "image/jpeg",
    "status": "CHANGED",
    "version": 2,
    "parent": 12386548974,
    "user": "Cristian",
    "client_modified": "2013-03-08 10:36:41.997",
    "server_modified": "2013-03-08 10:36:41.997"
  },
  {
    "name": "Winter2015.jpg",
    "path": "/documents/clients/Winter2015.jpg",
    "id": 32565632156,
    "size": 775412,
    "mimetype": "image/jpeg",
    "status": "RENAMED",
    "version": 3,
    "parent": 12386548974,
    "user": "Adrian",
    "client_modified": "2013-03-08 10:36:41.997",
    "server_modified": "2013-03-08 10:36:41.997"
  }
]
```


Get file version metadata

To retrieve information about a file version, an application submits an HTTP GET request to the file version resource.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with `/version/version-number`, for example, `https://domain.ext/stacksync/file/2148742318/version/2`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/file/2148742318/version/2
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.

Content-Type	The content type and character encoding of the response.
--------------	--

Response Body

The response body contains the associated metadata of the file in JSON:

ELEMENT	DESCRIPTION
name	The user-visible name of the fil.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
checksum	The file’s checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
{
  "name": "Winter2012.jpg",
  "path": "/documents/clients/Winter2012.jpg",
  "id": 32565632156,
```

```
"size":7482,  
"mimetype":"image/jpg",  
"status":"CHANGED",  
"version":2,  
"parent":12386548974,  
"user":"Cristian",  
"client_modified":"2013-03-08 10:36:41.997",  
"server_modified":"2013-03-08 10:36:41.997"  
}
```

Get file version data

To retrieve data about a file version, an application submits an HTTP GET request to the file version resource.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/file`, and ends with `/version/version-number/data`, for example, `https://domain.ext/stacksync/file/2148742318/version/2/data`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/file/2148742318/version/2/data
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content (if any).

Content-Type	The content type and character encoding of the response.
--------------	--

Response Body

The response body contains the retrieved file data.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248

<File content.... >
```

Folder

Create a folder

An application can create a folder by issuing an HTTP POST request to the URL of the containing folder resource. In addition, the application needs to provide as input, JSON that identifies the display name of the folder to be created.

Request

URL structure

The URL that represents the folder resource. The URL begins with `https://domain.ext/stacksync/folder`, for example, `https://domain.ext/stacksync/folder/8376125`.

Method

POST

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.
Content-Length	The length of the request body.
Content-Type	The content type and character encoding of the response. The content type must be <code>application/xml</code> , and the character encoding must be UTF-8.

Request Body

JSON input that contains a dictionary with the following keys:

ELEMENT	DESCRIPTION	REQUIRED
name	The user-visible name of the folder to be created.	Yes
parent	ID of the folder where the folder is going to be uploaded. If	No

	no ID is passed, it will use the top-level folder. This parameter should not point to a file.	
--	--	--

HTTP Request Example

```
POST https://domain.ext/stacksync/folder
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
Content-Length: 294
Content-Type: application/json
```

```
{
  "name": "clients",
  "parent": 681465736
}
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.
Location	The location of the newly created file.

Response Body

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
parent	ID of the folder's parent.

Response example

HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
Content-Length: 248
Location: <https://domain.ext/stacksync/folder/9873615>

```
{  
  "name": "clients",  
  "path": "/documents/clients",  
  "id": 9873615,  
  "parent": -348534824681,  
  "user": "Adrian"  
}
```


Delete a folder

An application can permanently delete a folder by issuing an HTTP DELETE request to the URL of the folder resource. It's a good idea to precede DELETE requests like this with a caution note in your application's user interface.

Caution: Deleting a folder means delete all its content.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/folder`, for example, `https://domain.ext/stacksync/folder/2148742318`.

Method

DELETE

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
DELETE https://domain.ext/stacksync/folder/2148742318
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content (if any).
Content-Type	The content type and character encoding of the response.

Response Body

The response body contains the associated metadata of the deleted file:

ELEMENT	DESCRIPTION
name	The user-visible name of the file.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
{
  "name": "clients",
  "path": "/documents/clients",
  "id": 9873615,
  "status": "DELETED",
  "version": 3,
```

```
"parent":-348534824681,  
"user":"Adrian",  
"client_modified":"2013-03-08 10:36:41.997",  
"server_modified":"2013-03-08 10:36:41.997"  
}
```

Get folder metadata

To retrieve information about a folder, an application submits an HTTP GET request to the folder resource that represents the folder.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/folder`, for example, `https://domain.ext/stacksync/folder/2148742318`.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/folder/2148742318
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.

Content-Type	The content type and character encoding of the response.
--------------	--

Response Body

The response body contains the associated metadata of the file in JSON:

ELEMENT	DESCRIPTION
name	The user-visible name of the file.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
status	Possible values are "NEW", "CHANGED", "RENAMED", "DELETED". Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder's parent.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=UTF-8
Content-Length: 248
```

```
{
  "name": "clients",
  "path": "/documents/clients",
  "id": 9873615,
  "status": "NEW",
  "version": 1,
  "parent": -348534824681,
  "user": "Adrian",
  "client_modified": "2013-03-08 10:36:41.997",
  "server_modified": "2013-03-08 10:36:41.997"
```

}

Get folder content metadata

To retrieve information about a folder, an application submits an HTTP GET request to the folder resource that represents the folder.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/folder`, followed by `/contents`, for example, `https://domain.ext/stacksync/folder/2148742318/contents`.

Parameters

The URL can be specified with the following optional query parameters:

ARGUMENT	VALUE	DESCRIPTION
include_deleted	true or false	<i>False</i> is the default. If this parameter is set to <i>true</i> , then response will include metadata of deleted objects.

Method

GET

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.

HTTP Request Example

```
GET https://domain.ext/stacksync/folder/2148742318/contents
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.

Response Body

The response body contains a JSON dictionary with the following keys:

ELEMENT	DESCRIPTION
folders	A list with all the folders.
files	A list with all the files.

Each folder metadata is composed by:

ELEMENT	DESCRIPTION
name	The user-visible name of the file.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
status	Possible values are “NEW”, “CHANGED”, “RENAMED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example to determine if a file has changed or not.

server_modified	This is the modification time set by the server at the time of processing the file.
-----------------	---

Each file metadata is composed by:

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
size	The file size in bytes.
mimetype	The media type of the file. http://www.iana.org/assignments/media-types
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
checksum	The file’s checksum.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

????????????????

Update folder metadata (move and rename)

An application can update various attributes of a folder by issuing an HTTP PUT request to the URL that represents the folder resource. In addition, the app needs to provide as input, JSON that identifies the new attribute values for the folder. Upon receiving the PUT request, the StackSync service examines the input and updates any of the attributes that have been modified.

Here are the file attributes that can be updated:

- name
- parent

No other file attributes can be modified using a PUT request.

Note: An application can move a file to a different parent folder by changing the value of the parent element.

Request

URL structure

The URL that represents the file data resource. The URL begins with `https://domain.ext/stacksync/folder`, and ends with the file ID, for example, `https://domain.ext/stacksync/folder/2148742318`.

Method

PUT

Request Headers

The request header includes the following information:

FIELD	DESCRIPTION
StackSync-API	API version. The value must be v2.
Content-Length	The length of the request body.
Content-Type	The content type and character encoding of the response. The content type must be <code>application/xml</code> , and the character encoding must be UTF-8.

Request Body

JSON input that contains a dictionary with the following keys:

ELEMENT	DESCRIPTION	REQUIRED
name	The user-visible name of the folder.	No
parent	ID of the folder where the folder is going to be moved. If parent is None the folder will be moved to the root folder.	No

HTTP Request Example

```
POST https://domain.ext/stacksync/folder/2148742318
X-Auth-Token: 657AD54T651ZE68745
StackSync-API: true
Content-Length: 294
Content-Type: application/json
```

```
{
  "name": "clients_2",
  "parent": 12386548974
}
```

Response

Response Header

The response header includes the following information:

FIELD	DESCRIPTION
Content-Length	The length of the retrieved content.
Content-Type	The content type and character encoding of the response.
Location	The location of the newly created file.

Response Body

ELEMENT	DESCRIPTION
name	The user-visible name of the file to be created.
path	The absolute path to the file.
id	A unique identifier for a file or folder.
status	Possible values are “NEW”, “CHANGED”, “DELETED”. Indicating the status of the file in this specific version.
version	A unique identifier for the current version of a file. Can be used to detect changes and avoid conflicts.
parent	ID of the folder’s parent.
client_modified	This is the modification time set by the desktop client when the file was last modified, in the standard date format. Since this time is not verified (the server stores whatever the desktop client sends up), this should only be used for display purposes (such as sorting) and not, for example, to determine if a file has changed or not.
server_modified	This is the modification time set by the server at the time of processing the file.

Response example

HTTP/1.1 200 OK
 Content-Type: application/json; charset=UTF-8
 Content-Length: 248
 Location: https://domain.ext/stacksync/file/

```

{
  "name": "Winter2012_renamed.jpg",
  "path": "/documents/clients/Client1.pdf",
  "id": 32565632156,
  "status": "CHANGED",
  "version": 2,
  "parent": 12386548974,
  "user": "Adrian",
  "client_modified": "2013-03-08 10:36:41.997",
  "server_modified": "2013-03-08 10:36:41.997"
}

```