

Rediafile

Cloud API v2

Developer & Business Guide

Complete integration guide for developers and business customers.
Uploading, managing and sharing files via REST API.

Base URL	https://rediafile.com/cloud/api/v2/
Format	JSON / UTF-8
Protocol	HTTPS (recommended)
Method	POST (unless otherwise stated)
Date	August 2026

Table of Contents

- 1. Overview
- 2. Quick Start
- 3. Authentication
 - 3.1 /authorize
 - 3.2 /disable_access_token
- 4. HTTP Error Codes
- 5. Endpoints Account
 - 5.1 /account/info
 - 5.2 /account/package
- 6. File Endpoints
 - 6.1 /file/upload
 - 6.2 /file/download
 - 6.3 /file/url_upload_add
 - 6.4 /file/url_upload_status
 - 6.5 /file/info
 - 6.6 /file/edit
 - 6.7 /file/delete
 - 6.8 /file/move
 - 6.9 /file/copy
- 7. Endpoints Folders
 - 7.1 /folder/create
 - 7.2 /folder/listing
 - 7.3 /folder/info
 - 7.4 /folder/edit
 - 7.5 /folder/delete
 - 7.6 /folder/move
- 8. Code Examples
- 9. Best practices
- 10. Limits & Restrictions
- 11. FAQ

1. Overview

The Rediafile API allows you to securely integrate file uploading, management, and downloading. directly from your own applications. All endpoints return responses in JSON format. Encoded in UTF-8.

Guaranteed compatibility — This API will evolve over time, but no current endpoint or response value will not be modified. You can write your integration code with confidence.

Authentication flow: 1. Authenticate (/authorize) ÿ 2. Get Token ÿ 3. Call API ÿ 4. JSON Response

Base URL:

<https://rediafile.com/cloud/api/v2/>

Technical requirements:

Appearance	Specification
Encoding	UTF-8 is required for all channels.
Protocol	HTTPS is strongly recommended (SSL required).
HTTP method	POST (unless otherwise stated)
Date format	Ymd H:i:s (ex: 2026-08-01 14:12:11)
Content-Type	application/x-www-form-urlencoded or multipart/form-data

2. Quick Start

Make your first API call in 3 steps:

Step 1 — Retrieve your API keys

Log in to your Rediafile account › Account Settings › API Access. You will find Key 1 and Key 2 there (64 characters each).

Step 2 — Obtain an access_token

```
curl -X POST https://rediafile.com/cloud/api/v2/authorize \
-d "username=your_username" \ -d
"password=your_password"
```

Step 3 — Use the token

```
curl -X POST https://rediafile.com/cloud/api/v2/account/info \
-d "access_token=YOUR_TOKEN" \ -d
"account_id=YOUR_ACCOUNT_ID"
```

3. Authentication

All API requests require a valid `access_token` and `account_id`. These are obtained via the endpoint `/authorize`.

Session — The same access token can be reused multiple times. Do not generate a new token for Each request. Tokens expire automatically after 1 hour of inactivity.

3.1 POST /authorize

Provides an `access_token` and an `account_id` to perform subsequent queries.

<https://rediafile.com/cloud/api/v2/authorize>

Parameters:

Setting	Kind	Required	Description
username	string	Yes*	Account username
password	string	Yes*	Account password
key1	string (64)	Yes*	API key 1 (from Account Settings)
key2	string (64)	Yes*	API key 2 (from Account Settings)

* Either username/password or key1/key2. API keys are recommended for integrations. automated.

Successful response:

```
{
  "data": {
    "access_token": "X3Xp6cUcue22Q3AlpCiZz3mJQWPT2v10zZqGbISGzViqZiMoV4ou8LeYH4SKAUL9...",
    "account_id": "158642"
  },
  "_status": "success",
  "_datetime": "2026-08-01 14:12:11"
}
```

Possible errors:

Message	Cause
Please provide a username.	Missing username parameter
Please provide a password.	Missing password setting
Please provide key1. It must be 64 characters.	Invalid API key 1
Please provide key2. It must be 64 characters.	Invalid API key 2
Could not authenticate user.	Invalid login details or locked account
Failed to issue access token.	Internal generation error

3.2 POST /disable_access_token

Disables an active `access_token`. Optional — tokens expire automatically after 1 hour of inactivity.

https://rediafile.com/cloud/api/v2/disable_access_token

Parameters: access_token (required), account_id (required)

4. HTTP Error Codes

Errors are returned using standard HTTP codes. Additional information is included.
in the body of the response in JSON format.

Code	Description	Action
400	Bad input parameter	Check the parameters sent.
401	Bad or expired token	Re-authenticate yourself via /authorize
404	File not found	Check the file_id or short_url
405	Method not expected	Use POST (or GET if specified)
429	Rate limit reached	Reduce the frequency of requests
5xx	Server error	Try again later or contact support

JSON error format:

```
{  
  "status": "error",  
  "response": "Could not authenticate user...",  
  "_datetime": "2026-08-01 14:12:11"  
}
```

5. Endpoints Account

5.1 POST /account/info

Provides details of the account associated with the account_id.

<https://rediafile.com/cloud/api/v2/account/info>

Parameters: access_token (required), account_id (required)

Answer :

```
{
  "data": {
    "id": "158642",
    "username": "admin",
    "level_id": "20",
    "email": "email@yoursite.com",
    "lastlogindate": "2017-02-18 11:43:39",
    "lastloginip": "192.168.33.1",
    "status": "active",
    "title": "Mr",
    "firstname": "Admin",
    "lastname": "User",
    "languageid": "1",
    "datecreated": null,
    "lastPayment": "2011-12-27 13:45:22",
    "paidExpiryDate": null,
    "storageLimitOverride": null
  },
  "_status": "success",
  "_datetime": "2026-08-01 14:12:11"
}
```

5.2 POST /account/package

Provides account restrictions inherited from the associated plan.

<https://rediafile.com/cloud/api/v2/account/package>

Key fields in the answer:

Field	Description
label	Name of the package (e.g., Premium Account)
max_upload_size	Maximum upload size in bytes
can_upload	1 = upload allowed, 0 = not
concurrent_uploads	Maximum number of simultaneous uploads
max_remote_download_urls	Maximum remote URLs (50 by default)
level_type	Account type (paid user, etc.)

6. File Endpoints

6.1 POST /file/upload

File upload interface. Note: Chunk uploads are not yet supported.

<https://rediafile.com/cloud/api/v2/file/upload>

Parameters:

Setting	Kind	Required	Description
access_token	string	Yes	Access token
account_id	string	Yes	Account ID
upload_file	file	Yes	File to upload (multipart/form-data)
folder_id	string	No	Target folder ID (root if empty)

Content-Type required — For this endpoint, use multipart/form-data with the attached file.

Successful response:

```
{
  "response": "File uploaded",
  "data": {
    "name": "sample4_l.jpg",
    "size": "149084",
    "type": "application/octet-stream",
    "url": "https://rediafile.com/2Vv",
    "delete_url": "https://rediafile.com/2Vv-d?41efa710...",
    "info_url": "https://rediafile.com/2Vv-i",
    "stats_url": "https://rediafile.com/2Vv-s",
    "short_url": "2Vv",
    "file_id": "1253",
    "unique_hash": "60b0be7e3b18de9a3f00d940a8e5a983...",
    "url_html": "&lt;a href=...&gt;view sample4_l.jpg on Rediafile&lt;/a&gt;",
    "url_bbcode": "[url]https://rediafile.com/2Vv/[url]"
  },
  "_status": "success",
  "_datetime": "2026-08-01 14:12:11"
}
```

6.2 POST /file/download

Generates a unique download URL for a file.

<https://rediafile.com/cloud/api/v2/file/download>

Parameters: access_token, account_id, file_id OR short_url (one of the two is required)

Answer :

```
{
  "data": {
    "file_id": "1253",
    "filename": "sample4_l.jpg",
    "download_url": "https://rediafile.com/2Vv?download_token=c3e6289a..."
  },
  "_status": "success",
  "_datetime": "2026-08-01 14:12:11"
}
```

6.3 POST /file/url_upload_add

Imports a file from an external URL. The URL must be public (not password protected).

https://rediafile.com/cloud/api/v2/file/url_upload_add

Parameters:

Setting	Kind	Required	Description
access_token	string	Yes	Access token
account_id	string	Yes	Account ID
file_url_base64	string	Yes	External URL encoded in base64
folder_id	string	No	Target folder ID

Answer :

```
{
  "response": "File download from URL scheduled",
  "data": {
    "scheduled_item_id": 52,
    "url": "https://example.com/image.png"
  },
  "_status": "success"
}
```

6.4 POST /file/url_upload_status

Displays the status of a program URL download.

https://rediafile.com/cloud/api/v2/file/url_upload_status

Parameters: access_token, account_id, scheduled_item_id

Response (status complete):

```
{
  "response": "Scheduled URL file download found",
  "data": {
    "scheduled_item_id": 51,
    "status": "complete",
    "created": "2024-01-25 19:34:52",
    "started": "2024-01-25 20:24:53",
    "finished": "2024-01-25 20:24:54",
    "total_size": 17262,
    "downloaded_file": {
      "file_id": 11456,
      "name": "logo.png",
      "size": 17262,
      "url": "https://rediafile.com/sZG/logo.png"
    }
  },
  "_status": "success"
}
```

6.5 POST /file/info

Provides the metadata and URLs of a file.

<https://rediafile.com/cloud/api/v2/file/info>

Parameters: access_token, account_id, file_id

6.6 POST /file/edit

Modifies the metadata of a file.

<https://rediafile.com/cloud/api/v2/file/edit>

Parameters:

Setting	Kind	Required	Description
access_token	string	Yes	Access token
account_id	string	Yes	Account ID
file_id	string	Yes	ID of the file to be modified
filename	string	No	New name (empty = unchanged)
fileType	string	No	New MIME type
folder_id	string	No	New parent folder (empty = unchanged)

6.7 POST /file/delete

Deletes an active file (moves it to the recycle bin).

<https://rediafile.com/cloud/api/v2/file/delete>

Parameters: access_token, account_id, file_id

6.8 POST /file/move

Moves a file to another folder.

<https://rediafile.com/cloud/api/v2/file/move>

Parameters: access_token, account_id, file_id, new_parent_folder_id

6.9 POST /file/copy

Copy a file to another folder.

<https://rediafile.com/cloud/api/v2/file/copy>

Parameters: access_token, account_id, file_id, copy_to_folder_id

Answer :

```
{
  "response": "File successfully copied.",
  "original_file": { "data": { "id": "1162", "shortUrl": "2U2" } },
  "new_file": { "data": { "id": "1254", "shortUrl": "2Vw" } },
  "_status": "success"
}
```

7. Endpoints Folders

7.1 POST /folder/create

Create a new folder.

<https://rediafile.com/cloud/api/v2/folder/create>

Parameters:

Setting	Kind	Required	Description
access_token	string	Yes	Access token
account_id	string	Yes	Account ID
folder_name	string	Yes	New folder name
parent_id	string	No	Parent folder ID
is_public	int	No	0=Private, 1=Not listed, 2=Public (default: 0)
access_password	string(32)	No	MD5 hash of the password

7.2 POST /folder/listing

Returns the list of folders and files in a parent folder.

<https://rediafile.com/cloud/api/v2/folder/listing>

Parameters: access_token, account_id, parent_folder_id (optional, empty = root)

Answer :

```
{
  "data": {
    "folders": [ { "id": "123", "folderName": "My Folder 1" } ],
    "files": [ { "id": "1161", "filename": "button_back.gif", "shortUrl": "2U1" } ]
  },
  "_status": "success"
}
```

7.3 POST /folder/info

Provides information from a specific case.

<https://rediafile.com/cloud/api/v2/folder/info>

Parameters: access_token, account_id, parent_folder_id

7.4 POST /folder/edit

Modifies an existing file.

<https://rediafile.com/cloud/api/v2/folder/edit>

Parameters: access_token, account_id, folder_id, folder_name (opt), parent_id (opt), is_public (opt), access_password (opt)

7.5 POST /folder/delete

Delete a folder.

<https://rediafile.com/cloud/api/v2/folder/delete>

Parameters: access_token, account_id, folder_id

7.6 POST /folder/move

Moves a folder to another location.

<https://rediafile.com/cloud/api/v2/folder/move>

Parameters: access_token, account_id, folder_id, new_parent_folder_id

8. Code Examples

cURL — Authentication

```
curl -X POST https://rediafile.com/cloud/api/v2/authorize \-H "Content-Type: application/x-www-form-urlencoded" \-d "username=admin" \-d "password=my_password"
```

cURL — File Upload

```
curl -X POST https://rediafile.com/cloud/api/v2/file/upload \-F "access_token=YOUR_TOKEN" \-F "account_id=YOUR_ACCOUNT_ID" \-F "upload_file=@/path/to/myfile.zip" \-F "folder_id=123"
```

PHP — Full Upload

```
&lt;?php
$token = "YOUR_ACCESS_TOKEN"; $accountId = "YOUR_ACCOUNT_ID"; $filePath = "/path/to/file.jpg";

$postData = [
    'access_token' => $token, 'account_id' => $accountId, 'upload_file' => new CURLFile($filePath)
];

$ch = curl_init("https://rediafile.com/cloud/api/v2/file/upload"); $postData = json_encode($postData);

curl_setopt($ch, CURLOPT_POST, true); curl_setopt($ch, CURLOPT_POSTFIELDS, $postData); curl_setopt($ch, CURLOPT_RETURNTRANSFER, true); $response = curl_exec($ch); $data = json_decode($response, true);

echo "URL: " . $data["data"][0]["url"];
curl_close($ch); ?>
```

Python — Download from URL

```
import requests, base64

url = "https://rediafile.com/cloud/api/v2/file/url_upload_add" payload = {
    'access_token': "YOUR_TOKEN", 'account_id': "YOUR_ACCOUNT_ID", 'file_url_base64':
    base64.b64encode(b"https://example.com/image.png").decode(), 'folder_id': "123"
}

response = requests.post(url, data=payload) print(response.json())
```

JavaScript (Node.js) — List of folders

```
const axios = require('axios');

async function listFolders() {
    const res = await axios.post('https://rediafile.com/cloud/api/v2/folder/listing', { access_token: 'YOUR_TOKEN', account_id: 'YOUR_ACCOUNT_ID', parent_folder_id: "" });
}
```

```
    console.log(res.data.data.folders);  
  }  
  listFolders();  
}
```

JavaScript (Fetch) — Download link

```
fetch('https://rediafile.com/cloud/api/v2/file/download', {  
  method: 'POST',  
  headers: { 'Content-Type': 'application/x-www-form-urlencoded' }, body: new URLSearchParams({  
  
    access_token: 'YOUR_TOKEN', account_id:  
    'YOUR_ACCOUNT_ID', file_id: '1253'  
  
  })  
  
}).then(r => r.json()).then(data  
=> console.log(data.data.download_url));
```

9. Best practices

Security — Always use HTTPS

All requests must go through HTTPS. You can enforce SSL on the server side to ensure the security of tokens and data.

Performance — Reuse tokens

An access token remains valid for 1 hour. Store it in the session and reuse it for all your requests instead of regenerating a token for each call.

Reliability — Manage errors

Always check `_status` in the JSON response. If there is an error, read the `response` field for the detailed message.

Encoding — Base64 for URLs

For `/file/url_upload_add`, always encode the URL in base64 before sending it. Use utf-8 for all text.

Limitation — No chunked upload

Chunk uploads are not yet supported. For large files, use `/file/url_upload_add` or check `max_upload_size` via `/account/package`.

10. Limits & Restrictions

Appearance	Value / Behavior
Token expiration	1 hour of inactivity
Rate limiting	HTTP 429 if limit reached
Upload chunks	Not currently supported
Upload remote URL	Maximum 50 URLs (depending on the plan)
Encoding	UTF-8 required
API keys	64 characters exactly
Folder password	MD5 hash, 32 characters

Business plans — Storage, upload size and bandwidth limits depend on your plan.
Check </account/package> to see your current restrictions.

11. FAQ

Question	Answer
How do I obtain my API keys?	Log in to your Rediafile account › Account Settings › API Access. You will see
Can I use username/password instead of API keys?	Yes, /authorize accepts either username+password or key1+key2. API keys are rec
My token has expired, what should I do?	Call /authorize again to obtain a new access_token. Tokens expire
How do I upload to a specific folder?	Use the `folder_id` parameter in `/file/upload`. Get the folder ID via `/folder/listing`.
Is uploading large files supported? Chunk uploads	are not yet available. The maximum file size depends on your plan.
Can I integrate Rediafile into my mobile app?	Yes, the API is compatible with all platforms. Use HTTP POST requests.

Support

For any technical questions or support requests:

- Email: support@rediafile.com
- Online documentation: <https://rediafile.com/cloud/api>
- Developer portal: <https://developers.rediafile.com>