

How to automate data upload/download with the REST API

- [Purpose](#)
- [Using the REST API](#)
 - [SAMPLE Request](#)
- [Upload](#)
 - [Suggestion 1:](#)
 - [Suggestion 2:](#)
- [FAQ](#)
- [Download](#)
 - [Download full dataset](#)

Purpose

Data upload (and download) can be done using various mechanisms :

- Web-based interface at <https://staging-dataupload.wmda.info> (please note the URL for production is dataupload.wmda.info)
- Desktop applications and [Sync Clients for Mac, Linux, Windows](#).
- REST API, using commands like [cURL](#), or libraries like [Restsharp](#)

This document describes the data upload mechanism via the REST API method, where users can upload a PGP encrypted XML file.

Important note on security

The examples below are for educational purposes: WMDA

Please do read <https://ec.haxx.se/usingcurl-netrc.html> on the use of passwords and usernames if you plan to automate these procedures.

```
# example use:
curl --netrc-file mycredentialsfile https://xyz.com

# the mycredentialsfile contains :
machine <xyz.com> login <johndoe@xyz.com> password <uf9873o^9ufwa>
```

WMDA dataupload production plan to do a maintenance and update between 08 Nov 2022 - 18 Nov 2022 , which will introduce security control as the update in staging from Oct, 27th, 2022. We will notify the date when we have the registries better test the new script to upload and download the file. From staging update we know this will impact the current method used in the REST API. Check the "Upload" user guide below in upload suggestions

Using the REST API

To test the API you may use a tool like [Restlet](#), a plugin for Chrome browsers to test API calls before you deploy. The server responds with a HTTP code 200 in when succesful, but the response body may contain detailed error messages.

In case the upload succeeded, an XML response with meta information of the upload is generated. You may use this for own logging purposes. For more info you can read the [full API specification](#).

URL	
SERVER URL	Staging system: https://staging-dataupload.wmda.info Production system: https://dataupload.wmda.info

URI	/api/v2/io/ION1234/ Explanations: ▪ Please refer to the file naming convention. This is stated as : For XML format: The files are marked with ION numbers (ION-1234-D or ION-1234-C) ▪ The ION number is the unique ION number of an organisation sending the file. ▪ The D give the information that it is a donor file and the C give the information that it is a cord blood file
METHOD	PUT
Full URL path example	For donor for organization with ION-1234, the full path is as below: https://staging-dataupload.wmda.info/api/v2/io/ION1234/ION-1234-D.gpg
HEADER	Authorization and Content-Type are required, the other ones are optional
Authorization	basic
Content-Type	application/octet-stream (when your file < 128K) multipart/form-data (when your file >128K) From 27 Oct 2022 , Content-Type is required. See our suggestion below in "Upload".
cX-Rename-If-Exists	true

- Server name should be <https://staging-dataupload.wmda.info>
- For production you will need <https://dataupload.wmda.info>

SAMPLE Request

First test the connection by using the right path and fetching the metadata : it should result in a response 200 OK.

- Look carefully at the path : here we use *ion0999*, that should be replaced by your ION.
- You may notice that **we add the filename to the path**: depending on the library you may need to add that

DRAFT

Save as ▼

METHOD
PUT

SCHEME // HOST [**:PORT] [PATH ["?" QUERY]]
https://dataupload.wmda.info/api/v2/io/ION0999/ion-0999-c.xml.gpg
length: 65 bytes

Send ▼

▼ QUERY PARAMETERS
+ Add query parameter

HEADERS ⓘ
☒ Authoriz : Basic aXRAbWF0

+ Add header

+ Add authorization

Form

BODY ⓘ

ion-0999-c.xml.gpg
1012 bytes, drop

File ▼

Upload

Now we know the path is correct and we can do an upload using cURL:

Below is the invalid to deploy, currently it is still available in PRODUCTION till production maintenance. But not available any more in STAGING dataupload.

Invalid after update from Oct, 27, 2022

```
curl -i -X PUT -H "Authorization:Basic d21.....uZT==" -T "/test.pgp" 'https://staging-dataupload.wmda.info/api/v2/io/ION0999/'
```

Below are suggestions with more secure headers:



The curl script needs update to upload in STAGING from Oct, 27th, 2022 after maintenance/update . And the new suggested script can be used in PRODUCTION as well. So we strongly suggest API users to update, then there will be less stress after PRODUCTION update /maintenance.

For registries use other script, please test in STAGING as well, and adjust with the Content-Type if not used before. We noticed that the uploaded file format can be binary or with info to identify it is PGP/GPG MESSAGE. And for PGP/GPG MESSAGE format, there are more solutions.

Suggestion 1:

For all the files, script as below is suggested for file larger than 128K.

Curl for linux

```
curl -i -X PUT -H "Authorization:Basic d21.....uZT==" -H "Content-Type: multipart/form-data" --data-binary "@path/to/file/ION-0999-D.gpg" "https://staging-dataupload.wmda.info/api/v2/io/ION0999/" -H "X-File-Name:ION-0999-D.gpg"
```

Curl for windows

```
curl -i -X PUT -H "Authorization:Basic d21.....uZT==" -H "Content-Type: multipart/form-data" --data-binary "@c:/path/to/file/ION-0999-D.gpg" "https://staging-dataupload.wmda.info/api/v2/io/ION0999/" -H "X-File-Name:ION-0999-D.gpg"
```

Suggestion 2:

For files that encrypted and identified as PGP/GPG MESSAGE (open the encrypted file and you can see this info), then -F and -H options also work. -F already included "Content-Type: multipart/form-data", so no need to provide it any more.

For binary file, extra header information will be added to the content of the file when file is uploaded, and then it can not be decrypted any more, so please use suggestion 1 to upload.

Curl for linux

```
curl -i -X PUT -H "Authorization:Basic d21.....uZT==" -F "file=@path/to/file/ION-0999-D.gpg" "https://staging-dataupload.wmda.info/api/v2/io/ION0999/" -H "X-File-Name:ION-0999-D.gpg"
```

Curl for linux

```
curl -i -X PUT -H "Authorization:Basic d21.....uZT==" -F "file=@c:/path/to/file/ION-0999-D.gpg" "https://staging-dataupload.wmda.info/api/v2/io/ION0999/" -H "X-File-Name:ION-0999-D.gpg"
```

FAQ

Question : My report states : File could not be processed due to a file decryption error, when using the API

Answer :

The Dataupload's API endpoint /io streams file(s) directly to the workspace. We have detected that premature file pickups may occur and attempt to move/delete files while content is still being appended or processed. We have taken actions to reduce this behaviour by decreasing the frequency for file pickup which in turn does lower the chance for premature pick. We have also implemented gpg error validation to check integrity based on its messages.

Note: Files are picked from the workspaces for processing every 10 minutes.

Some key messages:

- Known internal gpg messages for corrupted files during API uploads: "invalid packet", "invalid encoding" and "failed".
- Report message that uploader users would receive in case of corrupted gpg file upload: File could not be processed due to a file decryption error. Please make sure file is properly encrypted.

For more information please contact WMDA support team.

Download

Download reports

You may use the API also to fetch your reports . Unfortunately, it is not one line cmd, and need 2 steps.

1.Fetch the files list and get the file names.

Use the following curl and endpoint to fetch an array with filedescriptors: url should use double quote or no quote in windows env:

```
curl -H "Authorization:Basic d21....uZT==" https://staging-dataupload.wmda.info/api/v2/fs/reports-ion0999/?children=f
```

You will get the result in xml format, and you can get json format, use the cmd below:

```
curl -H "Authorization:Basic d21....uZT==" https://staging-dataupload.wmda.info/api/v2/fs/reports-ion0999/?children=f&format=json
```

2. Fetch all the reports or the one you need.

Note, you need to use "io" instead of "fs" in the path.

```
curl -O GET -H "Authorization:Basic d21....uZT==" 'https://staging-dataupload.wmda.info/api/v2/io/reports-ion0999/$filename' -o '$filename.txt'
```

In windows cmd, the -o dose not work, and the cmd is as below:

```
curl -H "Authorization:Basic d21....uZT==" https://staging-dataupload.wmda.info/api/v2/io/reports-ion0999/$filename > $filename.txt
```

Download archived files

Download archived files will be the same as download reports. And 2 steps as above are needed.

The example URL for archive folder is as below:

```
https://staging-dataupload.wmda.info/api/v2/fs/archive-ion0999/?children=f
```

In RESTLET :

RESPONSECache Detected - Elapsed Time: 1.39s

200 OK

HEADERS [?]

pretty ▾

pre>Date: 2017 Oct 12 12:43:45
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips PHP/7.1.5
Vary: Authorization
X-Powered-By: PHP/7.1.5
Connection: close
Content-Length: 515 bytes
Content-Type: text/xml; charset=UTF-8
COMPLETE REQUEST HISTORY

pretty ▾

BODY [?]

pretty ▾

<?xml version="1.0" encoding="UTF-8" ?>
<tree>
 <nodes_diff>
 <add>
 <tree is_file="true" filename="/ION0999" mimeType="application/json" ajxp_description="Modified today at 10:43" />
 </add>
 </nodes_diff>
</tree>

length: 515 bytes

Click on Code to get raw CURL syntax:

Save Code

da.info/api/v2/io/ION0999/

Response should be 200:

RESPONSECache Detected - Elapsed Time: 1.39s

200 OK

HEADERS [?]

pretty ▾

pre>Date: 2017 Oct 12 12:43:45
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips PHP/7.1.5
Vary: Authorization
X-Powered-By: PHP/7.1.5
Connection: close
Content-Length: 515 bytes
Content-Type: text/xml; charset=UTF-8
COMPLETE REQUEST HISTORY

pretty ▾

BODY [?]

pretty ▾

<?xml version="1.0" encoding="UTF-8" ?>
<tree>
 <nodes_diff>
 <add>
 <tree is_file="true" filename="/ION0999" mimeType="application/json" ajxp_description="Modified today at 10:43" />
 </add>
 </nodes_diff>
</tree>

length: 515 bytes

Download full dataset

For those registries that are permitted to use the full dataset the statement below will fetch that. Please change the ION workspace to match your ION.

```
curl -O GET -H "Authorization:Basic ....." "https://dataupload.wmda.info/api/v2/io/downloads/ION1804/bmdw4data.zip.gpg" -o 'wmda_data_v22.zip.gpg'
```

In windows cmd, the -o dose not work, and the cmd is as below:

```
curl -H "Authorization:Basic ....." "https://dataupload.wmda.info/api/v2/io/downloads/ION1804/bmdw4data.zip.gpg" > wmda_data_v22.zip.gpg
```

We noticed sometimes the download maybe terminated because of some temp network limitation, and the "-C -" option can be used to continue the download.

-v is for more details of the track info

```
curl -v -H "Authorization:Basic ....." -o 'wmda_data_v22.zip.gpg' -C - 'https://dataupload.wmda.info/api/v2/io/downloads/ION1804/wmda_data_v22.zip.gpg'
```