API for TrueSigner ONE

(API for Multi-signer usecases using Aadhaar / DSC /Electronic signing)

Truecopy Credentials Pvt Ltd
helpdesk@truecopy.com

Introduction

a) This API is used to trigger a file for signing by one or more signatories, using either Aadhaar Signatures or DSC signatures or Electronic Signatures.

b) The API involves essentially uploading a PDF document, and specifying the signatories, their email ids / mobile numbers, along with other parameters such as callback url, signature locations etc.

c) There are two flavours of this API. There is a JSON version and there is a multi-part version. You can use either of the two, depending on what you are comfortable with. Both APIs do exactly the same thing.

d) Once you trigger a document for signing, it is sent to the specified signatories for their signatures (either on email or SMS as the case may be). When a signatory signs a document, you will receive a call-back informing you of this fact. The callback URL is to be specified by you.

e) Finally, when the file is signed by all signatories, you have an API to download the final signed document signed by all signatories. (Note: You cannot download a partially signed document – say signed by 3 out of 8 signatories, because that violates the interests of the 3 signatories who have signed it)

Deployment and API Parameters

a) You will be assigned a unique subdomain for production and for UAT, and hence all calls / urls that you andyour signers use will be of the form: *https://yoursubdomain.truecopy.in/.......*
When your subdomain is created, you will be assigned a secret API-KEY and an initiator account, which you can use for authenticating the API-calls.

b) Each API call is made with a unique identifier (called UUID) to be specified by you. You have to create this UUID, and ensure that it is unique for each file that you initiate for signing.The UUID should contain alphanumerics and should ideally be about 32 characters. The UUID is used to identify a particular file by you and the Truecopy-Gateway.

c) You can create workflows in your account to specify the parameters that don’t change for each of your calls, such as signature type (Aadhaar / DSC / Esign etc), the size of the signature and so on.

API Calls

1A) Upload PDF File For Signing (json version)

POST Request

POST */services/corpservice/v2/uploadfilejson* HTTP/1.1

Host: subdomain.truecopy.in
Content-Type: json
Accept: application/json
Body:

```
 {
“uuid”:”<uuid to be specified by you>”,
“cs”:”<checksum value>”,
“initiator”:”<initiator id provided to you>”,
“signerlist”:”<signerlist in format specified>”,
“workflow_id”:”<workflow_id_value>”,
“callbackurl”:”<callbackurl_value>”,    //optional
“uploadfile”:”<B64 encoded content of the pdf file>”
 }
```

DESCRIPTION:

uuid = unique id for each file to be specified by the user. Usually a 32 char alphanumeric string

cs = security checksum computed as: sha256(APIKEY+uuid)
cs is thus the sha256 hash in hex format of the string formed by concatenation of your APIKEY and the specific uuid for that file. This means the cs differs for each uuid.

initiator = initiator account created for you.

signerlist = signer info json string.

e.g.

```
[{"signer":"a@b.com","signer_name":"AB","signer_mobile":"9xxxxxxx12","signloc”
:"1[72:102],1[383:559]","seq_no":1,"parallel_index":1},{"signer":"e@f.com","signer
_name":"EF",“signer_mobile":"9xxxxxxx34”,"signloc":"2[32:12],1[33:59]","seq_no"
:2,"parallel_index":1},{"signer":"m@n.com","signer_name":"MN","signer_mobile":"
9xxxxxxx56”,"signloc":"1[72:102],1[33:55]","seq_no":2,"parallel_index":2}]
```

This is a signer list for 3 signers, where signer a@b.com signs first (seq_no = 1) and after he has signed signers e@f.com and m@n.com both can sign in any sequence (both have same seq_no).

The signing URLs are sent to the email IDs specified in the “signer” field for each signer. They can also be sent by SMS to the phone number specified for the signer in the optional field “signer_mobile”

The respective signature locations are specified in the format P[X:Y] where P is the page no, and X,Y are the respective coordinates on that page.

e.g. #1
"1[50:200],4[300,500]" will put the signature at X=50, Y=200 on page 1 and also at X=300, Y=500 on page 4.
e.g.#2
"[10,20]" will put the signature at X=10, Y=20 on all pages.

workflow_id = id of the workflow created in your portal account for one-time definition of several parameters such as sign type, size, etc.

callbackurl = base url which will be used for callback with GET parameters appended by TrueSigner (This parameter is optional and can also be configured via your portal account for your subdomain)

uploadfile = B64 encoded string of contents of the pdf file to be uploaded for signing

Response

Http/1.1 200 OK (on Success)

Server: subdomain.truecopy.in
Content-Type: application/json
Content-Length: {length}
Date: {optional}

The following data is returned:

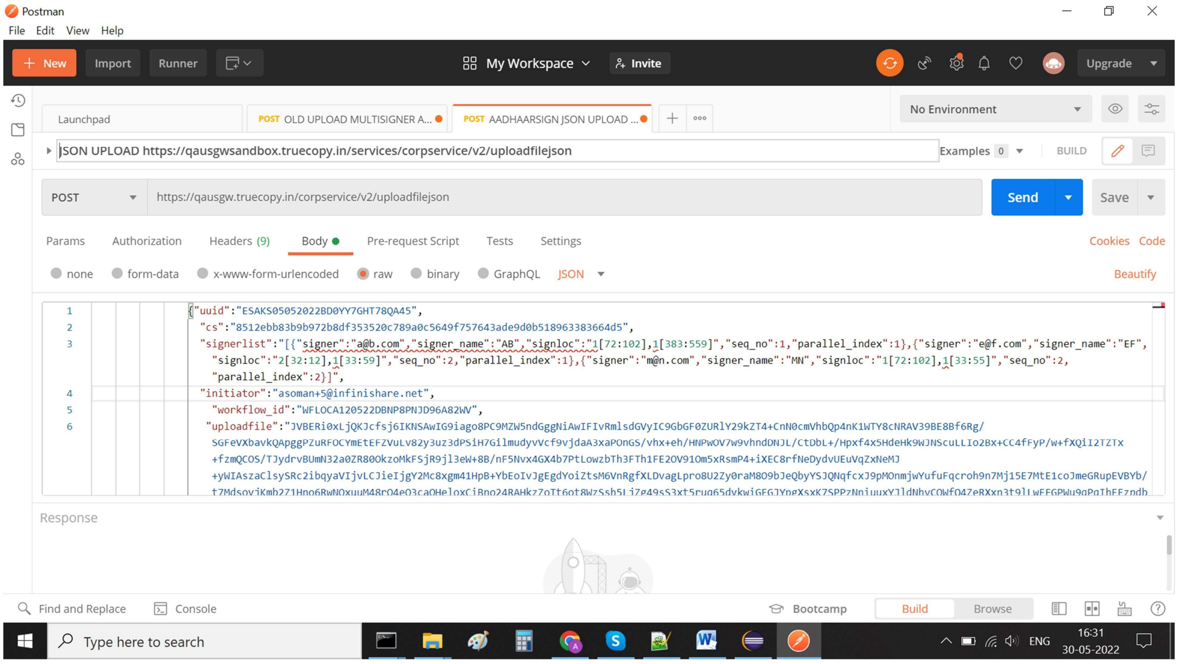
```
{
    "status": 0,
    "errCode": "",
    "message": "File successfully uploaded",
    “url”: “Signing URL” //Only if it’s a single Aadhaar signer
    "uuid":  "DSCAKS05052022BD0YY7GHT78LO22"
}
```

The URL is returned only if it’s a Single signer scenario with Aadhaar / DSC signing. This URL displays a signing page / widget that can be used to embed into your Portal.

Http/1.1 500 (For error)

{"status":<error code>,,"message":"<ERROR MESSAGE>"}

POSTMAN SCREENSHOT:



JSON UPLOAD https://qausgwsandbox.truecopy.in/services/corpservice/v2/uploadfilejson

POST https://qausgw.truecopy.in/corpservice/v2/uploadfilejson

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{"uuid":"ESAKS05052022BD0YY7GHT78QA45",
  "cs":"8512ebb83b9b972b8df353520c789a0c5649f757643ade9d0b518963383664d5",
  "signerlist":"[{"signer":"a@b.com","signer_name":"AB","signloc":"1[72:102],1[383:559]","seq_no":1,"parallel_index":1},{"signer":"e@f.com","signer_name":"EF",
    "signloc":"2[32:12],1[33:59]","seq_no":2,"parallel_index":1},{"signer":"m@n.com","signer_name":"MN","signloc":"1[72:102],1[33:55]","seq_no":2,
    "parallel_index":2}]",
  "initiator":"asoman+5@infinishare.net",
    "workflow_id":"WFLOCA120522DBNP8PNJD96A82WV",
   "uploadfile":"JVBERi0xLjQKJcfsj6IKNSAwIG9iago8PC9MZW5ndGggNiAwIFIvRmlsdGVyIC9GbGF0ZURlY29kZT4+CnN0cmVhbQp4nK1WTY8cNRAV39BE8Bf6Rg/
    SGFeVXbavkQApggPZuRFOCYmEtEFZVuLv82y3uz3dPSiH7GilmudyvVcf9vjdaA3xaPOnGS/vhx+eh/HNPwOV7w9vhndDNJL/CtDbL+/Hpxf4x5HdeHk9WJNScuLLIo2Bx+CC4fFyP/w+fXQiI2TZTx
    +fzmQCOS/TJydrvBUmN32a0ZR80OkzoMkFSjR9jl3eW+8B/nF5Nvx4GX4b7PtLowzbTh3FTh1FE2OV91Om5xRsmP4+iXEC8rfNeDydvUEuVqZxNeMJ
    +yWIAszaClsySRc2ibqyaVIjvLCJieIjgY2Mc8xgm41HpB+YbEoIvJgEgdYoiZtsM6VnRgfXLDvagLpro8U2Zy0raM8O9bJeQbyYSJQNqfcxJ9pMOnmjwYufuFqcroh9n7Mj15E7MtE1coJmeGRupEVBYb/
    t7MdsovjKmb27lHno6RwNOxuuM48rO4eO3caOHeloxCjBno24RAHkz7oTt6ot8WzSsh5LjZg49sS3xt5rug65dvkwiGFGJYngXsxK7SPPzNnjuuxYJldNhvCOWfO47eRXxn3t9lLwFFGPWu9gPgIhFFznd
```

1B) Upload PDF File For Signing (multipart-form version)

POST Request

```
POST /services/corpservice/v2/uploadfile      HTTP/1.1

Host: subdomain.truecopy.in
Content-Type: multipart/form-data, boundary=----------AaBbCc
Accept: application/json

----------AaBbCc
Content-Disposition: form-data; name="uuid"
<uuid_value>
----------AaBbCc
Content-Disposition: form-data; name="cs"
<cs_value>
----------AaBbCc
Content-Disposition: form-data; name="initiator"
<initiator_id>
----------AaBbCc
Content-Disposition: form-data; name="workflow_id"
<workflow_id_value>
----------AaBbCc
Content-Disposition: form-data; name="callbackurl"
<callbackurl_value>
----------AaBbCc
Content-Disposition: form-data; name="signerlist"
<signerlist_value>
----------AaBbCc
Content-Disposition: form-data; name="uploadfile"
Content-Type: application/pdf
<file-contents (binarydata)>
----------AaBbCc
```

Response

Http/1.1 200 OK (on Success)
Server: subdomain.truecopy.in
Content-Type: application/json
Content-Length: {length}
Date: {optional}

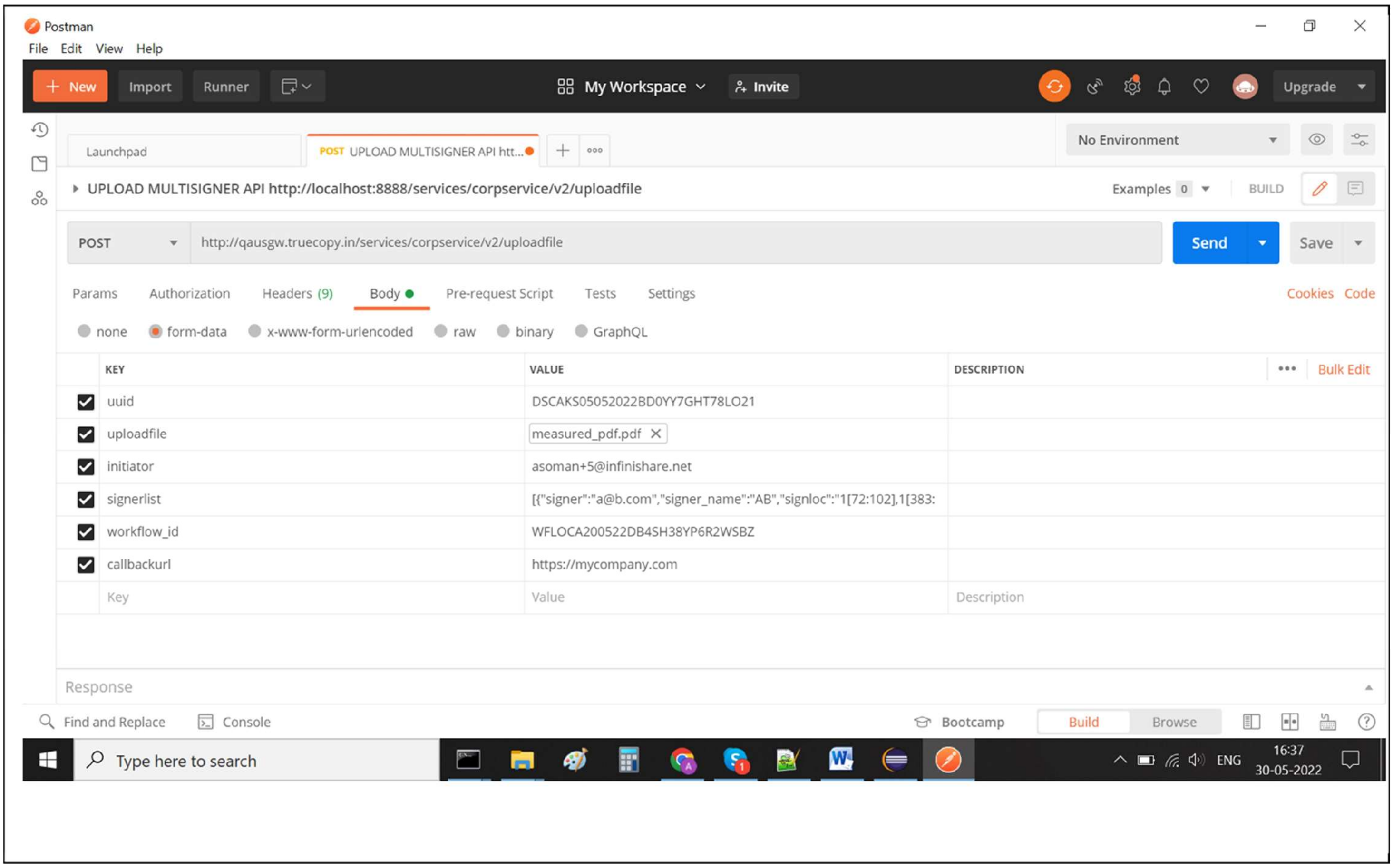
The following data is returned:

```
{
    "status": 0,
    "errCode": "",
    "message": "File successfully uploaded",
    “url”: “Signing URL” //Only if it’s a single Aadhaar or DSC signer
    "uuid":  "DSCAKS05052022BD0YY7GHT78LO22"
}
```

The URL is returned only if it’s a Single signer scenario with Aadhaar / DSC signing. This URL displays a signing page / widget that can be used to embed into your Portal.

Http/1.1 500 (For error)
{"status":<error code>,,"message":"<ERROR MESSAGE>"}

POSTMAN Screenshot:



UPLOAD MULTISIGNER API http://localhost:8888/services/corpservice/v2/uploadfile

POST http://qausgw.truecopy.in/services/corpservice/v2/uploadfile

○ none ● form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	KEY	VALUE	DESCRIPTION
☑	uuid	DSCAKS05052022BD0YY7GHT78LO21	
☑	uploadfile	measured_pdf.pdf ✕	
☑	initiator	asoman+5@infinishare.net	
☑	signerlist	[{"signer":"a@b.com","signer_name":"AB","signloc":"1[72:102],1[383:	
☑	workflow_id	WFLOCA200522DB4SH38YP6R2WSBZ	
☑	callbackurl	https://mycompany.com	
	Key	Value	Description

2. Callback after user signs:

Callback is a GET request made to the callback URL provided.

When the document is signed by any signer, Truecopy will do a GET request (callback) to the ***callbackURL*** provided by you with GET parameters as below:

*https://**callbackURL**?uuid=<uuid of the file>&cs=<checksum computed for that uuid>&status=<signing status>&info=<additional information>&mi=<b64 value>&grp=<groupstatus>*

From the UUID in the callback you can know which document the call back pertains to. Based on the values of other GET parameters you can then proceed to take a suitable action.

The GET parameters returned will be as follows:

uuid = the UUID of the document / data that was signed.

cs = checksum computed available to ensure that the callback indeed came from Truecopy

status = SIGNED (when successful)
FAILED (when signing fails)

info = information about error returned by ESP (if any)

mi = Base64 encoded Signer information provided by ESP in case of Aadhaar signatures such signer name, last 4 digit of Aadhaar number, Address Pincode as per Aadhaar, Year of Birth etc

grp = If there are multiple signers, this parameter shows grp=COMPLETE after final signatory has signed the document.

3. Download Signed File

GET Request
GET */services/corpservice/v2/fetchsignedfile/{uuid}/{cs}* HTTP/1.1 **Host:** subdomain.truecopy.in **Accept:** */*
GET parameters **uuid** = unique id for the file (to be specified by the partner) **cs** = same checksum as used during upload
Response **Http/1.1 200 OK** Server: subdomain.truecopy.in Content-Type: application/octet-stream Content-Length: {length} Date: {optional} == binary document data (signed pdf file)== **OR** **500 In case of error or if the file is not available**

ERROR CODES

65	CHECKSUM_MISMATCH
67	UUID_ALREADY_EXISTS
69	INCORRECT_DATA
75	ERROR_SAVING_FILE
63	UNKNOWN_ERROR
78	SIGN_COUNT_EXCEEDED
998	INVALID_AADHAAR_NO
91	ACCESS_DENIED
66	RESOURCES_NOT_FOUND
404	FILE_NOT_FOUND