



Enabling Trust

Remote identification service integration

Technical description

2018

Contents

- 1 Document..... 3
- 2 Process schema 3
- 3 Client’s requirements..... 3
- 4 HTTP REST communication 4
- 5 Process Example 7
- 6 Status check API with email 9
- 7 Technical workflow of identification statuses 10

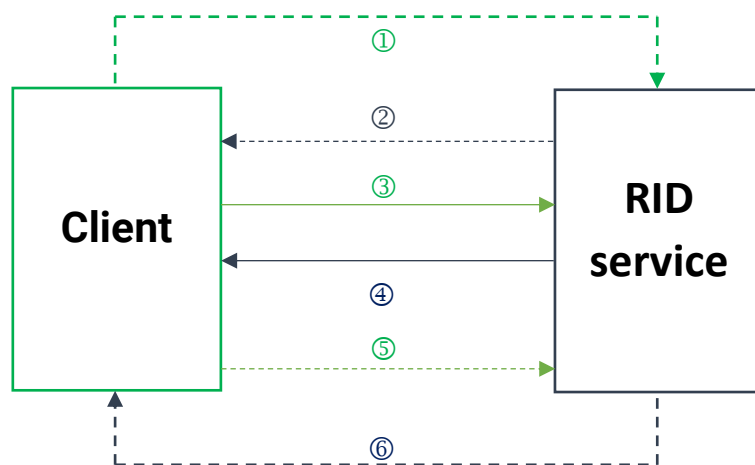
1 Document

This document provides a technical specification of WEB integration of Identification service in client WEB page using iframe technology or simple redirect function.

Definitions

- **RID** stands for Remote Identification
- **RID service** is a system which helps to provide identification service.
- **Client** is a company that is using RID service.
- **Client's customer (Customer)** is a Client's user

2 Process schema



———— Frontend process

- - - - - Backend process

Steps:

In case there is no active person identification in identification DB, Client initiate person identification process via RID Identification WEB

1. Client requests for token on server-side to start identification session for Customer;
2. RID service returns token for Client;
3. Client redirects Customer to RID service identification WEB using received token (GET or POST);
4. After identification completed or failed, RID service returns Customer to Client's WEB with token (GET).
5. Client requests identification data using token
6. RID service responds with identification data

After identification is completed, Client can request identification DB for person identification information

3 Client's requirements

For Company to be able to use the RID service, it is required to sign a service agreement. As a part of the registration process, the Company will receive parameters. Username, password which will be used in all communication with RID service for identification, verification and security purposes.

Client WEB should use HTTPS communication protocol that keeps any kind of data safe while in transit between Client server, RID service and Customer computer.

Client should provide to RID service:

- a) URL the Customer should be redirected after identification process;
- b) URL which should be requested in backend on identification status change.

4 HTTP REST communication

Steps:

1. Client send request with parameters below for token:

POST URL: [https://\[address\]/api/startsession](https://[address]/api/startsession)

Headers: Content-Type: application/json

POST BODY parameters:

No.	Field Name	Field Type	Field Description
1.	apikey	STRING	User API KEY
2.	data	ARRAY	Array of Customers data

Possible data elements:

Element	Type	Meaning
refId	String?, length min 1 max 255	Referral Id from client
name	String?, length min 2 max 255	Customer Name
surname	String?, length min 2 max 255	Customer Surname
email	String?, length min 5 max 255	Customer email
phoneNumber	String?, length min 6 max 22, symbols 0-9 /)(+.-	Customer phone number
nationality	String?, length 3	3-letter ISO ALPHA-3 Country Code (LTU)
birthday	String?, length 10, format yyyy-MM-dd	Birthday (yyyy-MM-dd)
personCode	String?, length min 5 max 64	Customer person code
personAddress	String?, length min 5 max 255	Customer living address
income	String? Decimal?, length min 1	Customer annual incomes
occupation	String?, length min 1 max 255	Customer annual incomes
source	String?, length min 3 max 255	Source of incomes
companyCode	String?, length min 5 max 255	Company Code
companyName	String?, length min 5 max 255	Company Name
legalForm	String?, length min 5 max 255	Company Legal Form
companyRegistrationAddress	String?, length min 5 max 255	Company Registration Address
relationshipToTheCompany	String?, length min 5 max 255	Relationship to the Company
activityInTheCompany	String?, length min 1 max 255	Activity in the company
beneficialOwners	String?, length min 5 max 255	Beneficial owner

Example:

```
{
  "apikey": "testapi",
  "data": {"email": "email@test.eml"}
}
```

- RID service store requests parameters and returns Token with RequestStatus. Token is valid until identification process is completed but not longer than half an hour.

Example:

```
{
  "token": "4593934b01814d65bb8ba4s9e0f65f90"
}
```

- After session started, Client navigates Customer to WEB identification page:
Navigation URL (using GET method):

URL: [https://\[address\]/Home/Start?token=#token](https://[address]/Home/Start?token=#token)

Default process language is English. Parameter should be added for other language:

LT - [https://\[address\]/lt/Home/Start?token=#token](https://[address]/lt/Home/Start?token=#token)

RU - [https://\[address\]/ru/Home/Start?token=#token](https://[address]/ru/Home/Start?token=#token)

EN - [https://\[address\]/en/Home/Start?token=#token](https://[address]/en/Home/Start?token=#token)

- After identification is completed or rejected, RID service returns Customer to URL given by the Client, using GET method: ClientsCallbackURL + „?token=#token

In case identification is confirmed or rejected manually via BackOffice, KYC service will send token to Clients URL in the same way using GET method:

BackendClientsCallbackURL + „?token=#token

- Client make request for token information via REST API (Request on token can be made within identification process)

POST URL: [https://\[address\]/api/getdata](https://[address]/api/getdata)

Headers: Content-Type: application/json

POST BODY parameters:

No.	Field Name	Field Type	Field Description
1.	apikey	STRING	User API KEY
2.	token	STRING	token

Example:

```
{
  "apikey": "testapi ",
  "token": "0792c588941f4d889a3fe45873dafa22"
}
```

- RID service returns information about identification.

Example:

```
{
  "status": 1,
  "data": {
    "name": "CustomerName",
    "surname": "CustomerSurnam",
    "email": " email@test.eml ",
    "nationality": null,
    "year": null,
    "month": null,
    "day": null,
    "personCode": null,
    "phoneNumber": ""
  }
}
"documentData": {
  "personCode": "39001010000",
  "birthDate": "1990-01-01",
  "firstName": "CustomerName",
  "lastName": "CustomerSurnam",
}
```

```

"documentNumber": "N15555555",
"documentType": 2,
"expireDate": "2019-08-18",
"country": "LTU",
"nationality": "LTU"
}

```

Possible statuses:

status	Meaning
0	Started
1	AutoFinish
2	ManualFinish
3	Failed
4	Expired
5	Rejected

Document data:

No.	Field Name	Field Type	Field Description
1.	personCode	STRING	Person Code
2.	birthDate	STRING	Person birth date
3.	firstName	STRING	Firs name
4.	lastName	STRING	Surname
5.	documentNumber	STRING	Document number
6.	documentType	STRING	Document type: 1 IdentityCard 2 Passport 3 ResidencePermit 4 DriverLicence 5 LocalPassport 6 SocialId 7 Other
7.	expireDate	STRING	Document expire date
8.	country	STRING	Country issued document
9.	nationality	STRING	Nationality / Citizenship

5 Process Example

[address] = kycico.finpass.eu

1. Token Request (Step 1 & 2)

The screenshot shows a REST client interface. The top bar indicates a POST request to `https://kycico.finpass.eu/api/startsession`. The 'Body' tab is selected, showing a JSON payload:

```
1 {
2   "apikey": "f66406834d4fc...",
3   "data": {
4     "email": "test@test.eu",
5     "name": "john",
6     "surname": "joshua",
7     "nationality": "AUS",
8     "birthday": "1992-05-02",
9     "walletAddress": "0x123f6...",
10    "amount": "0.0000"
11  }
12 }
```

Below the request, the 'Body' tab shows the response:

```
1 {
2   "token": "a5bea58d3..."
3 }
```

2. User navigation for identification process via browser (Step 3)

The screenshot shows a browser address bar with the URL `https://kycico.finpass.eu/Home/Start?token=a5bea58c...`

VIDEO AUTHENTICATION

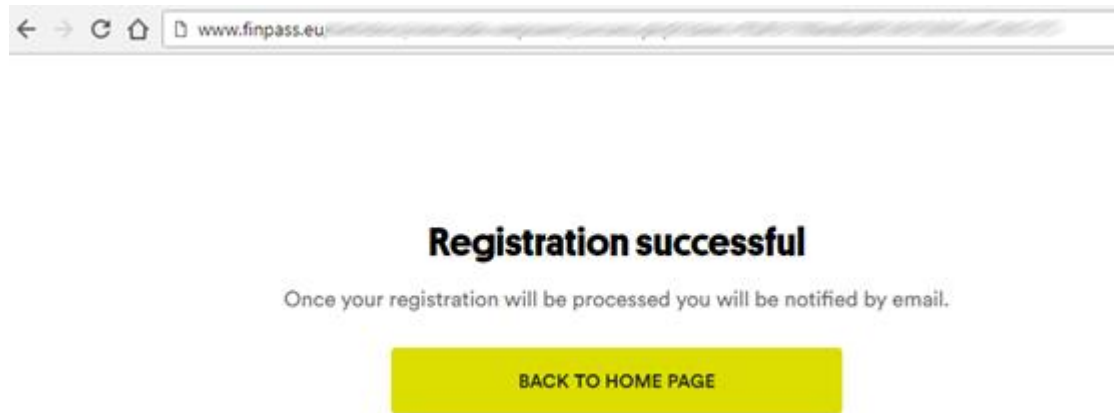
1 Record your face video



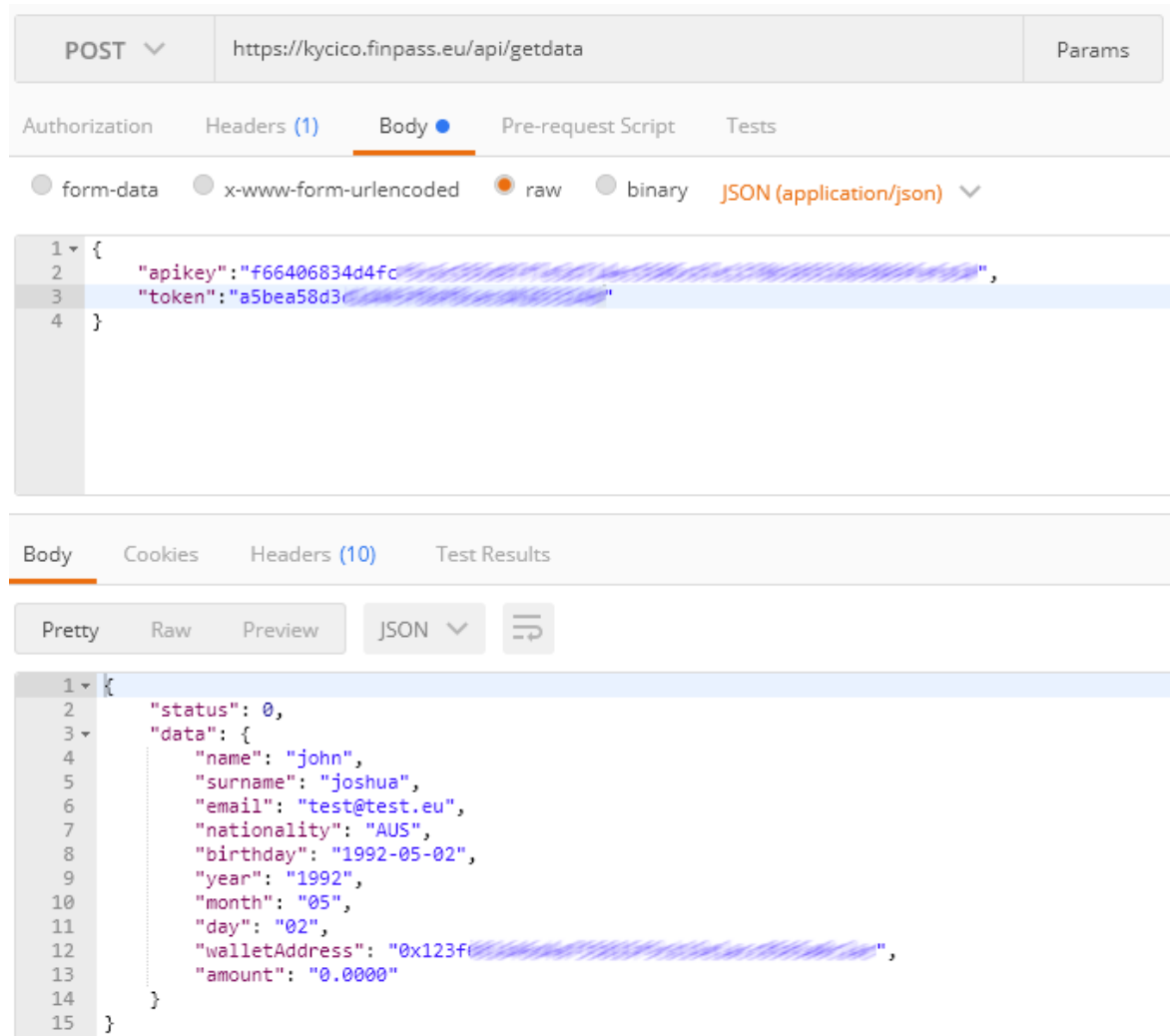
2 Take your ID or passport photo



3. User navigation back to client callback URL (Step 4)



4. Gathering identification status and form data information (Step 5 & 6)



6 Status check API with email

POST URL: [https://\[address\]/api/registrationStatus](https://[address]/api/registrationStatus)

Headers: Content-Type: application/json

Request

a) Request with Email

```
{
  "apiKey": "string",
  "email": "string",
}
```

No.	Field Name	Field Type	Field Description
1.	apikey	STRING	Client API KEY
2.	email	STRING	Customer email

Response

```
{
  "StatusId": "int",
  "IdentificationStatusText": "string",
  "RejectReasonId": "int",
  "RejectReasonText": "string"
}
```

* !! NULL values not returned.

No.	Field Name	Field Type	Field Description
1.	StatusId	STRING	Identification status Id table "Possible statuses"
2.	IdentificationStatusText	STRING	Identification status name
3.	RejectReasonId	STRING	Rejection reason ID: table "Rejection reasons"
4.	RejectReasonText	STRING	Rejection reason name

Rejection reasons:

RejectReasonId	RejectReasonText	Suggestion
1	Missing document photo.	Please include document photo that contains your photo and general information
2	Document not accepted	Please use a nationally issued identity document with your photo and MRZ parameters (ID card or passport)
3	Data in the document does not match data in the application	Please make sure you entered data correctly
4	Name, nationality and date of birth was matched in at least one of the sanctions lists	Please contact us for assistance.
5	Missing individual's photo/video selfie.	Please submit your photo/video selfie.
6	Individual's face in the selfie does not match the face in the document.	Please make sure you used your own document.
7	Poor photo quality.	Please use a better camera to submit your photos.
8	Poor lighting, photo is too dark.	Please make sure there is sufficient lighting when taking a photo.

9	Blurred photo.	Please, make sure that information in the document is clearly visible prior to submitting it.
10	Face photo upload format.	Uploaded face photo doesn't contain the passport/ID card and a note with the "Client Name" and upload date.
11	Miscellaneous.	Please repeat identification.

7 Technical workflow of identification statuses

1. When initiating the identification process, an identifier (token) is created with the status 0 - StartSession;
The token can be created with API StartSession method.
2. If the Identification is not performed within 30 minutes, the assigned token becomes inactive, status 4 - Expired;
3. If a person submits media data for identification, one of the following statuses will be assigned:
 - a) AutoFinish (1) - If the data is high-quality, the photo of the document's face matches the face image, MRZ data is readable, not listed in sanction list.;
 - b) Failed (2) - If we do not specify the document in the data provided, or the face or the face does not match the photo of the document, or the face photo and document face image is very similar according to biometrics. (It is very likely that two identical pictures have been submitted, i.e. either only with the face or only with the document)

User will be redirected to thank you page (CallbackURL) provided with the token using GET method, for example:

<https://www.yourdomain.co/thankyou.html?token=3cd2bb1132044a2d9f7e6de7ecf1cb33>

After the token received, it should be requested via API [getdata] for status, form and document information.
4. After the automatic process, identifications with the statuses Failed (3) are reviewed via BO. Such identifications are reviewed and verified or rejected manually:
 - a) If confirmed - status changes to Manualfinish (2) - background callback is executed on client's system;
 - b) If rejected, the status changes to Rejected (5) - the background callback is executed on client's system;

The callback would be the same type as mentioned before, only it will be initiated at backend

<https://www.yourdomain.co/thankyou.html?token=3cd2bb1132044a2d9f7e6de7ecf1cb33>

After the token received, it should be requested via API [getdata] for status, form and document information.