

Introducer API — Integration Guide

Prepared for: Reclaim My Money → Reclaim My Money

Overview

The Introducer API allows you to programmatically submit client data and receive a completion URL. The client then clicks the link to verify their identity, sign, and complete their claim. You receive webhook notifications as the claim progresses.

Getting Started

To begin integrating, you'll need the following from your account manager:

1. **API Key** — format: `aff_<prefix>_<random>` . Store this securely; it cannot be retrieved again after generation.
 2. **Brand ID** — the UUID identifying the brand you're submitting leads to.
 3. **Base URL** — the domain for API requests (e.g. `https://reclaimmymoney.mis-soldexpert.co.uk`).
-

Authentication

Include your API key in every request using one of:

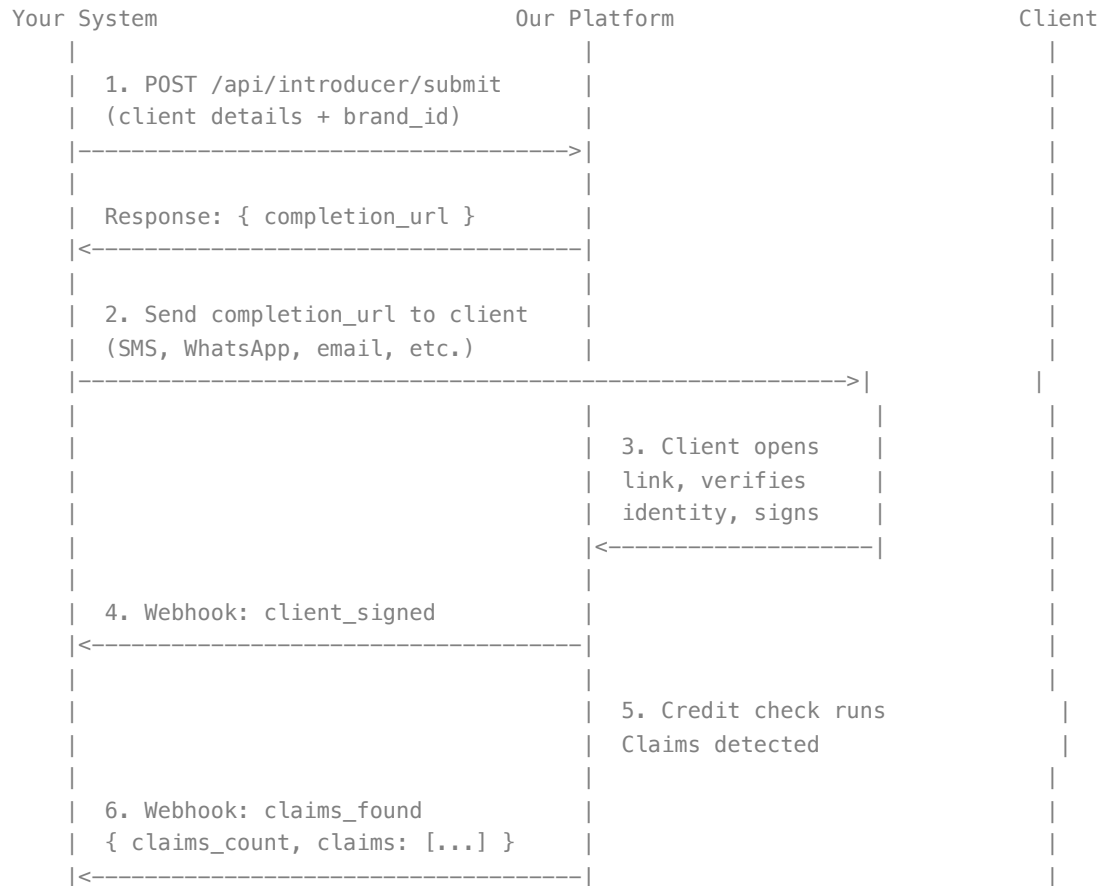
- `X-API-Key` header (preferred)
- `Authorization: Bearer <key>` header

Rate Limits

Limit	Default
Per hour	100 requests
Per day	1,000 requests

Exceeding limits returns `429 Too Many Requests` . Contact your account manager if you need higher limits.

Typical Flow



Steps 1–2 are your integration. **Steps 3–6** happen automatically.

Endpoints

POST /api/introducer/address/search

Search for UK addresses using a postcode or partial address.

Request:

```
{
  "search": "SW1A 1AA"
}
```

Response (200):

```
{
  "results": [
    {
      "id": "address_reference_id",
      "label": "42 High Street, London SW1A 1AA",
      "is_container": false
    }
  ],
  "session_id": "session_reference_id"
}
```

Notes:

- `is_container: true` means this is a group (e.g. a postcode covering multiple addresses). Re-search with the group's `id` as `session_id` to drill down to individual addresses.
 - `is_container: false` means this is a selectable address. Use its `id` with the retrieve endpoint or pass it directly as `address_id` in the submit request.
-

POST /api/introducer/address/retrieve

Retrieve full address details from an address reference ID.

Request:

```
{
  "address_id": "address_reference_id_from_search"
}
```

Response (200):

```
{
  "address": {
    "building_name": "",
    "house_number": "42",
    "flat": "",
    "street": "High Street",
    "town": "",
    "city": "London",
    "county": "Greater London",
    "postcode": "SW1A 1AA"
  }
}
```

The returned address object can be passed directly to the submit endpoint.

POST /api/introducer/submit

Submit client data and receive a completion URL.

Request:

```
{
  "brand_id": "a71eec45-e7cc-4101-9969-d0f5ddd5790b",
  "title": "Mr",
  "first_name": "John",
  "last_name": "Smith",
  "date_of_birth": "1985-03-15",
  "email": "john.smith@example.com",
  "phone": "07700900000",

  "address_id": "address_reference_id_from_search",

  "webhook_url": "https://your-system.com/webhooks/claims",
  "introducer_reference": "your-internal-ref-123",
  "send_sms": false
}
```

Required Fields

Field	Type	Description
brand_id	UUID	The brand to submit the lead to
title	string	Mr , Mrs , Miss , or Ms
first_name	string	Client's first name
last_name	string	Client's last name
date_of_birth	string	YYYY-MM-DD format
email	string	Valid email address
phone	string	UK mobile number
Address		Provide one of the two options below

Address Options

Option A (preferred): Provide `address_id` from the address search endpoint. The full address is resolved and validated automatically.

Option B: Provide an `address` object directly:

```

{
  "address": {
    "house_number": "42",
    "street": "High Street",
    "city": "London",
    "postcode": "SW1A 1AA",
    "building_name": "",
    "flat": "",
    "town": "",
    "county": "Greater London"
  }
}

```

`street`, `city`, and `postcode` are required. All other address fields are optional.

Optional Fields

Field	Type	Default	Description
<code>middle_name</code>	string		Client's middle name
<code>had_car_finance</code>	boolean		Whether the client has had car finance
<code>is_terms_policy_agreed</code>	boolean	<code>true</code>	Terms and conditions acceptance
<code>webhook_url</code>	string		URL to receive webhook events (overrides your default)
<code>introducer_reference</code>	string		Your internal reference — returned in webhooks and status responses
<code>send_sms</code>	boolean	<code>false</code>	Send an SMS to the client with the completion link
<code>agent_identifier</code>	string		Name or ID of the agent who handled the call
<code>utm_source</code>	string		UTM tracking parameter
<code>utm_medium</code>	string		UTM tracking parameter
<code>utm_campaign</code>	string		UTM tracking parameter
<code>custom_tracking</code>	object		Key-value pairs for custom tracking data

Response (201):

```

{
  "success": true,
  "claim_request_id": "550e8400-e29b-41d4-a716-446655440000",
  "claim_reference": "CL-A1B2C3D4",
  "completion_url": "https://reclaimmymoney.mis-soldexpert.co.uk/resume?token=...",
  "sms_sent": false,
  "crm_synced": true,
  "crm_skipped": false,
  "crm_client_uuid": "CUS-7",
  "action_points": [
    "Client must visit the completion_url to verify identity and sign."
  ]
}

```

Response Fields

Field	Type	Description
success	boolean	Whether the submission was accepted
claim_request_id	UUID	Unique identifier for this submission — use for status checks and deduplication
claim_reference	string	Human-readable claim reference
completion_url	string	Link for the client to verify identity and sign
sms_sent	boolean	Whether an SMS was sent to the client
crm_synced	boolean	Whether the client was successfully synced to the target CRM
crm_skipped	boolean	Whether CRM sync was intentionally skipped (some configurations defer sync until the client signs)
crm_client_uuid	string null	The client ID returned by the CRM, if sync occurred
action_points	string[]	List of next steps

CRM Sync

When a client is submitted, the system automatically syncs the client record to the brand's configured CRM. The `crm_synced`, `crm_skipped`, and `crm_client_uuid` fields tell you what happened:

- **crm_synced: true** — the client was created in the CRM at submit time. `crm_client_uuid` contains the CRM's client identifier.
- **crm_skipped: true** — CRM sync was intentionally deferred. Some brand configurations delay sync until a later step (e.g. after the client signs). The submission still succeeded — no action is needed from you.

- **Both false** — CRM sync was attempted but failed. The submission is aborted and returns an error (HTTP 422). In this case, do not send the completion URL to the client.

When the client later visits the completion URL and resumes the form, the system automatically re-syncs with the CRM as needed. No action is required from you.

What to Do with the Response

- **Save `claim_request_id`** — use it to check status later.
- **Send `completion_url` to the client** — via SMS (set `send_sms: true`), WhatsApp, email, or any channel you prefer.
- The client clicks the link, verifies their date of birth, signs the form, and their credit check runs automatically.

GET `/api/introducer/status/{claim_request_id}`

Check the status of a previously submitted claim request.

Response (200):

```
{
  "claim_request_id": "550e8400-e29b-41d4-a716-446655440000",
  "claim_reference": "CL-A1B2C3D4",
  "status": "pending",
  "introducer_reference": "your-internal-ref-123",
  "signed": false,
  "signed_at": null,
  "credit_check_completed": false,
  "claims_count": 0,
  "completion_url": "https://reclaimymoney.mis-soldexpert.co.uk/resume?token=...",
  "sms_sent": false,
  "sms_sent_at": null,
  "created_at": "2026-02-24T09:55:00.000Z"
}
```

Use this to poll for updates if you prefer polling over webhooks, or as a supplement to webhooks.

Webhooks

If you provide a `webhook_url` (either per-submission or configured as your default), you'll receive `POST` requests as the claim progresses.

`claims_found`

Fired when a credit check detects motor finance claims. Use this to track conversion and know whether the lead produced actionable claims.

Payload:

```
{
  "event": "claims_found",
  "claim_request_id": "550e8400-e29b-41d4-a716-446655440000",
  "claim_reference": "CL-A1B2C3D4",
  "timestamp": "2026-02-25T14:30:00.000Z",
  "introducer_reference": "your-internal-ref-123",
  "claims_count": 3,
  "claims": [
    {
      "lender_name": "Barclays",
      "account_type": "Motor Finance",
      "status": "PENDING"
    },
    {
      "lender_name": "Black Horse",
      "account_type": "Motor Finance",
      "status": "PENDING"
    }
  ],
  "client": {
    "first_name": "John",
    "last_name": "Smith",
    "email": "john.smith@example.com",
    "phone": "07700900000"
  }
}
```

The `claims` array provides a summary of each detected claim — lender name, account type, and current status.

client_signed

Fired when the client visits the completion URL and signs the form.

Payload:

```
{
  "event": "client_signed",
  "claim_request_id": "550e8400-e29b-41d4-a716-446655440000",
  "claim_reference": "CL-A1B2C3D4",
  "timestamp": "2026-02-24T11:00:00.000Z",
  "introducer_reference": "your-internal-ref-123",
  "claims_count": 3,
  "client": {
    "first_name": "John",
    "last_name": "Smith",
    "email": "john.smith@example.com",
    "phone": "07700900000"
  }
}
```

Webhook Delivery

- **Method:** POST with Content-Type: application/json
- **Timeout:** 10 seconds — your endpoint must respond within this window
- **Reliability:** All delivery attempts are logged. Webhook failures do not affect the client's claim flow.
- **Idempotency:** Your endpoint should handle receiving the same event more than once gracefully (use `claim_request_id + event` as a deduplication key).

Event Timing

The order of events depends on the claim flow:

Flow Type	Event Order
Credit-first	<code>claims_found</code> → <code>client_signed</code> (claims already detected at signing time, so <code>claims_count</code> is populated in both)
Standard	<code>client_signed</code> → <code>claims_found</code> (credit check runs after signing, so <code>claims_count</code> may be 0 in <code>client_signed</code>)

Your flow type is configured per brand — ask your account manager which applies to you.

Error Responses

All errors follow a consistent format:

```
{
  "error": "Human-readable error message"
}
```

Validation errors include field-level details:

```
{
  "error": "Validation failed",
  "fields": [
    { "field": "email", "message": "email must be a valid email address" },
    { "field": "date_of_birth", "message": "date_of_birth must be YYYY-MM-DD format" }
  ]
}
```

Status Codes

Code	Meaning
200	Success
201	Created (submit endpoint)
400	Bad request — validation error or missing required fields
401	Unauthorized — invalid, missing, or expired API key
403	Forbidden — inactive account or unauthorized brand access
404	Not found — invalid claim ID or address reference
429	Rate limit exceeded — wait and retry
500	Internal server error — contact support if persistent

Quick Start Example

A minimal integration in three steps:

1. Submit the client

```
curl -X POST https://reclaimmymoney.mis-soldexpert.co.uk/api/introducer/submit \
-H "X-API-Key: 0420458f..." \
-H "Content-Type: application/json" \
-d '{
  "brand_id": "a71eec45-e7cc-4101-9969-d0f5ddd5790b",
  "title": "Mr",
  "first_name": "John",
  "last_name": "Smith",
  "date_of_birth": "1985-03-15",
  "email": "john.smith@example.com",
  "phone": "07700900000",
  "address": {
    "street": "High Street",
    "house_number": "42",
    "city": "London",
    "postcode": "SW1A 1AA"
  },
  "webhook_url": "https://your-system.com/webhooks/claims",
  "introducer_reference": "your-ref-123"
}'
```

2. Send the link to your client

Use the `completion_url` from the response. Send it however you prefer — SMS, WhatsApp, email, or set `send_sms: true` to have us text it to the client automatically.

3. Receive webhooks

Your `webhook_url` will receive:

- **claims_found** — when claims are detected (includes lender names and count)
- **client_signed** — when the client signs the form

No further action is needed from you. The claim is now in progress.

Address Lookup (Optional)

If you have a postcode but not a full address, use the address endpoints before submitting:

```
# Step 1: Search by postcode
curl -X POST https://reclaimmymoney.mis-soldexpert.co.uk/api/introducer/address/search \
  -H "X-API-Key: 0420458f..." \
  -H "Content-Type: application/json" \
  -d '{"search": "SW1A 1AA"}'

# Step 2: Use the address_id in your submit request
curl -X POST https://reclaimmymoney.mis-soldexpert.co.uk/api/introducer/submit \
  -H "X-API-Key: 0420458f..." \
  -H "Content-Type: application/json" \
  -d '{
    "brand_id": "a71eec45-e7cc-4101-9969-d0f5ddd5790b",
    "title": "Mr",
    "first_name": "John",
    "last_name": "Smith",
    "date_of_birth": "1985-03-15",
    "email": "john.smith@example.com",
    "phone": "07700900000",
    "address_id": "id_from_search_results",
    "webhook_url": "https://your-system.com/webhooks/claims",
    "introducer_reference": "your-ref-456"
  }'
```

Using `address_id` is preferred as it guarantees a verified, correctly formatted address.

SMS Options

Approach	How	When to use
We send SMS	Set <code>"send_sms": true</code> in your submit request	Simplest option — we text the completion link to the client's mobile
You send the link	Set <code>"send_sms": false</code> (default), use the returned <code>completion_url</code>	When you want to control the messaging channel and content

Support

For integration support, API key provisioning, or rate limit adjustments, contact your account manager.