

PROOF OF ADDRESS IMPLEMENTATION GUIDELINES

Proof of Address Implementation Guidelines

Code:	
Version:	1.0
Date of version:	2024-Mar-07
Created by:	Martin Heikkilä, Head of Support
Approved by:	Jason Coombes, Head of Risk and Compliance
Confidentiality level:	PUBLIC

PUBLIC

Table of Contents

.....	1
Proof of Address Implementation Guidelines	2
API Overview.....	5
Request.....	6
Request Response.....	7
Upload document.....	8
Response Result.....	9
Errors.....	11
Coverage.....	13
Handwritten text:.....	13
Printed text:.....	13
Change History.....	15

PUBLIC

Proof of Address

This tool is designed for AML compliance, specifically for verifying users' addresses through Proof of Address (PoA) documents like Utility Bills and Bank Statements. It leverages advanced AI and machine learning models to extract, categorize, and structure data. This data is then matched against the customer's provided information, ensuring both accuracy and the document's timeliness.

Key Features:

- 1. Address Verification:** Ensures user's address matches with PoA documents.
- 2. PoA Document Upload:** Accepts Utility Bills or Bank Statements.
- 3. OCR Technology:** Converts text from images for data extraction.
- 4. AI-Driven Data Categorization:** Organizes extracted data effectively.
- 5. Fuzzy Matching:** Compares and contrasts the OCR results with pre-existing user details.
- 6. Issue Date Identification:** Determines the issuance date of PoA documents for recency verification.
- 7. Matching Accuracy:** Provides a percentage score of data matching.

This solution streamlines the process of address verification, enhancing both efficiency and compliance with AML standards.

PUBLIC

API Overview

Core Functionalities

This service enables users to upload a document to extract and verify name and address. To use this feature follow these steps:

1. Request: Initiate a POST request to the designated endpoint with the necessary parameters name and address.
2. Request Response: Receive a redirect URL in response.
 1. Access the given URL and upload the document which contains the name and address you want to verify. Accepted document types: **pdf, jpg and png** max size 20 MB
 2. After uploading the document, a message will appear indicating whether the document has been accepted or declined. If `redirect_success` and `redirect_failure` is set, the user will be redirected accordingly.
3. Response Result: Obtain the document scan results through a webhook or by initiating a GET request or use `redirect_success/redirect_failure` pages to handle results

Base URLs

- **Test Environment:** ``https://test-gateway.zignsec.com/core/api/``
- **Production Environment:** ``https://gateway.zignsec.com/core/api/``

Headers

Header	Description	Required
Authorization	This header parameter is the subscription key you received from ZignSec during the registration process. Example: Authorization: 123456add0cff22873c428e987654321	Yes
Content-Type	Specifies the media type of the request body data. Set to <code>application/json</code> if JSON object.	Yes

API - Documentation

- **Swagger UI:** [Swagger UI \(zignsec.com\)](https://swagger.zignsec.com)

API Endpoints

- **POST /api/sessions/proof_of_address/document**
 - Creates a new session with a redirect URL
- **GET /api/sessions/{id}**
 - Retrieves the current state of session, including any results or status updates.

Request

Sample POST Request

```
curl -X 'POST' \
'https://test-gateway.zignsec.com/core/api/sessions/proof_of_address/document' \
-H 'accept: application/json' \
-H 'authorization: Your-access-key' \
-H 'Content-Type: application/json' \
-H 'x-csrf-token: KjpOFAotcAA7fBIAU39SFFQ2ISZYAi4nZpvzekGxOJwBf-6PeyMy5dkl' \
-d '{
  "locale": "En",
  "metadata": {
    "first_name": "Martin",
    "full_address": " Kvarnvägen 4 17764 Järfälla ",
    "last_name": "Johansson"
  },
  "redirect_failure": "https://my_failure_url.com",
  "redirect_success": "https://my_success_url.com",
  "relay_state": "my-unique-customer-id",
  "webhook": "https://my_webhook_url.com"
}
```

Request Parameters

Parameter	Type	Description	Required
locale	string	Preferred Language to Use. Example: En	no
metadata	object	Metadata of the session	yes
first_name	string	The user's first name	yes
full_address	string	The user's complete address	yes
last_name	string	The user's last name	yes
redirect_failure	string	URL to redirect the end-user to on failure. Example: https://my_failure_url.com	no

Parameter	Type	Description	Required
redirect_success	string	URL to redirect the end-user to on success. Example: https://my_success_url.com	no
relay_state	string	Optional Custom Parameter. Example: my-unique-customer-id	no
webhook	string	Webhook URL where your backend will receive session events. Example: https://my_webhook_url.com	no

Request Response

Sample Response

```

{
  "data": {
    "id": "036758e7-89a4-4040-8478-8fb22e57a50b",
    "redirect_url": "https://test-gateway.zignsec.com/2fa-ui/poa/036758e7-89a4-4040-8478-8fb22e57a50b",
    "status": "pending"
  }
}

```

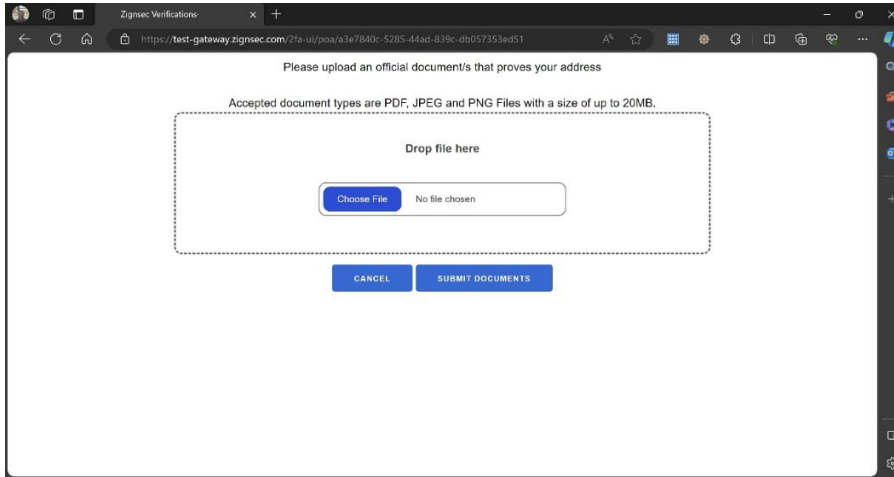
Response parameters

Parameter	Description
id	A unique session identifier generated for each workflow instance
errors	A JSON array of error conditions
redirect_url	The URL that the user needs to be redirected to upload the proof of address via the web interface.
status	Possible values are: Pending, submitted, accepted, declined, and cancelled.

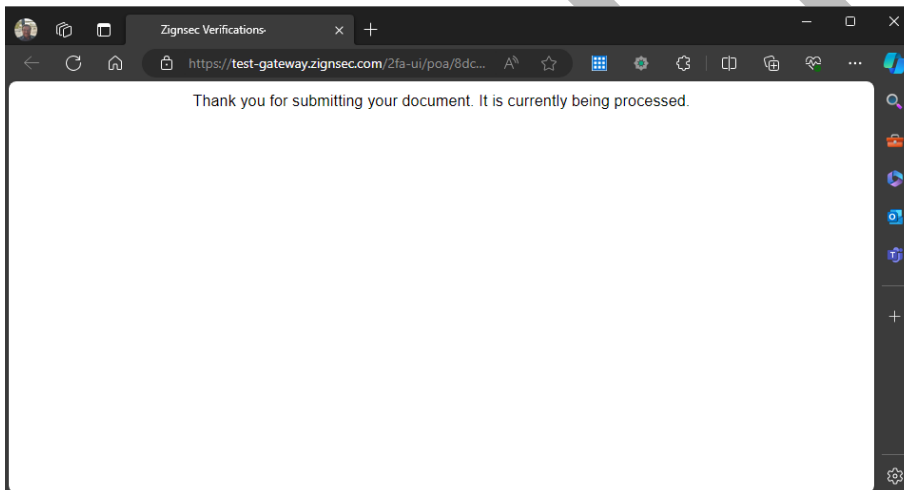
Upload document

Interface Snapshots Illustrating the Document Upload Feature.

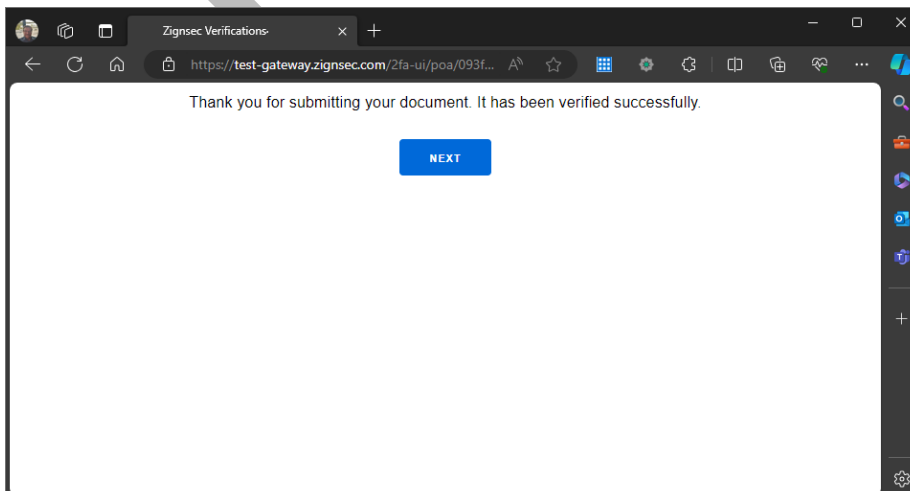
Step 1 Upload document



Step 2 Wait for document to be analyzed



Step 3 Document analyzed press next to continue to redirect set in Start request



Response Result

Sample GET Request

```
curl -X 'GET' \
'https://test-gateway.zignsec.com/core/api/sessions/f0254a90-da3f-440e-81fd-3c95d3577afc' \
-H 'accept: application/json' \
-H 'authorization: Your-Access-key'
```

Used to retrieve status and result response of the analysis. Simply pass the sessionID with the authorization token. (In example above sessionID is: f0254a90-da3f-440e-81fd-3c95d3577afc)

Sample Response after completing workflow

```
{
  "data": {
    "request_data": {
      "locale": "En",
      "metadata": {
        "first_name": "Martin",
        "full_address": "Kvarnvägen 4 17764 Järfälla",
        "last_name": "Johansson"
      },
      "redirect_failure": "https://my_failure_url.com",
      "redirect_success": "https://my_success_url.com",
      "relay_state": "my-unique-customer-id",
      "webhook": "https://webhook.site/zignsec"
    },
    "document_id": "f5b47938-f21a-42eb-ab40-787bbf4ad3f4",
    "first_name": "Martin",
    "folder_id": "e2bfd9b9-66ca-4b6a-86e2-84c3476f13cb",
    "full_address": "Kvarnvägen 4 17764 Järfälla",
    "id": "91ad1f95-48bb-427c-a736-69a13687412d",
    "last_name": "Johansson",
    "redirect_failure": "https://my_failure_url.com",
    "redirect_success": "https://my_success_url.com",
    "result": {
      "address_similarity": 1.0,
      "customer_address_recipient": "Martin Johansson",
      "date_validity_check": true,
      "extracted_address": "Kvarnvägen 4\n177 64 Järfälla",
      "extracted_address_structured": {
        "city": "Järfälla",
        "houseNumber": "4",
        "postalCode": "177 64",
        "road": "Kvarnvägen",
        "streetAddress": "4 Kvarnvägen"
      }
    }
  }
}
```

```

    },
    "extracted_name": "Martin Johansson",
    "invoice_date": "2023-12-07",
    "name_similarity": 1.0,
    "reason": "Passed: Address Similarity 1.0, Name Similarity: 1.0, Date Validity Check: Passed.",
    "recipient_similarity": 1.0,
    "vendor_name": "HSB®"
  },
  "status": "accepted"
}
}
    
```

Response parameters

Parameter	Type	Description
data	Object	Contains the response data
request_data	Object	Contains the json body sent in with the start request
document_id	String	Unique identifier for the document.
folder_id	String	Identifier for the folder.
id	String	Unique identifier of the data request used to GET the session information
result	Object	Contains the result of the data request
address_similarity	Number	Similarity score for address.
customer_address_recipient	String	Name of the customer at the address.
extracted_address	String	Extracted address from the document.
extracted_address_structured	Object	Structured form of the extracted address. <ul style="list-style-type: none"> "city": "Stockholm",

Parameter	Type	Description
		<ul style="list-style-type: none"> "houseNumber": "26", "postalCode": "123 45", "road": "Malmskillnadsgatan", "streetAddress": "26 Malmskillnadsgatan Lgh 1003", "unit": "Lgh 1003"
extracted_name	String	Extracted name from the document.
invoice_date	String	Date of the invoice as extracted from the document.
name_similarity	Number	Similarity score for the name.
reason	String	Summary of the checks done. Example: "Passed: Address Similarity 1.0, Name Similarity: 1.0, Date Validity Check: Passed.", "Date Validity Check: Invoice Date not within acceptable time range"
recipient_similarity	Number	Similarity score for the recipient.
vendor_name	String	Name of the vendor as extracted from the document.
status	String	Status of the request, e.g., 'accepted'.

Errors

Sample ERROR

```

{
  "detail": "Malformed payload",
  "instance": "/core/api/sessions/proof_of_address/document",
  "status": 400,
  "title": "Bad Request",
  "type": "about:blank",
  "violations": [
    "metadata/first_name :null value where string expected"
  ]
}
    
```

```
]
}
```

Errors

Parameter	Type	Description
detail	string	Error detail. Might contain technical information.
instance	string(\$uri)	A URI reference that identifies the problem type
status	string(\$uri)	The HTTP status code generated by origin server for this occurrence of the problem.
title	string	Error description. Human readable text.
type	string(\$uri)	A URI reference that identifies the problem type

PUBLIC

Coverage

Below is the list of the languages covered by the service.

Handwritten text:

Languages	
English	Japanese
Chinese Simplified	Korean
French	Portuguese
German	Spanish
Italian	

Printed text:

Languages	
Afrikaans	Khasi
Albanian	K'iche'
Angika (Devanagiri)	Korean
Arabic	Korku
Asturian	Koryak
Awadhi-Hindi (Devanagiri)	Kosraean
Azerbaijani (Latin)	Kumyk (Cyrillic)
Bagheli	Kurdish (Arabic)
Basque	Kurdish (Latin)
Belarusian (Cyrillic)	Kurukh (Devanagiri)
Belarusian (Latin)	Kyrgyz (Cyrillic)
Bhojpuri-Hindi (Devanagiri)	Lakota
Bislama	Latin
Bodo (Devanagiri)	Lithuanian
Bosnian (Latin)	Lower Sorbian
Brajbha	Lule Sami
Breton	Luxembourgish
Bulgarian	Mahasu Pahari (Devanagiri)
Bundeli	Malay (Latin)
Buryat (Cyrillic)	Maltese
Catalan	Malto (Devanagiri)
Cebuano	Manx
Chamling	Maori
Chamorro	Marathi
Chhattisgarhi (Devanagiri)	Mongolian (Cyrillic)
Chinese Simplified	Montenegrin (Cyrillic)
Chinese Traditional	Montenegrin (Latin)

Cornish	Neapolitan
Corsican	Nepali
Crimean Tatar (Latin)	Niuean
Croatian	Nogay
Czech	Northern Sami (Latin)
Danish	Norwegian
Dari	Occitan
Dhimal (Devanagiri)	Ossetic
Dogri (Devanagiri)	Pashto
Dutch	Persian
English	Polish
Erzya (Cyrillic)	Portuguese
Estonian	Punjabi (Arabic)
Faroese	Ripuarian
Fijian	Romanian
Filipino	Romansh
Finnish	Russian
French	Sadri (Devanagiri)
Friulian	Samoan (Latin)
Gagauz (Latin)	Sanskrit (Devanagiri)
Galician	Santali(Devanagiri)
German	Scots
Gilbertese	Scottish Gaelic
Gondi (Devanagiri)	Serbian (Latin)
Greenlandic	Sherpa (Devanagiri)
Gurung (Devanagiri)	Sirmauri (Devanagiri)
Haitian Creole	Skolt Sami
Halbi (Devanagiri)	Slovak
Hani	Slovenian
Haryanvi	Somali (Arabic)
Hawaiian	Southern Sami
Hindi	Spanish
Hmong Daw (Latin)	Swahili (Latin)
Ho(Devanagiri)	Swedish
Hungarian	Tajik (Cyrillic)
Icelandic	Tatar (Latin)
Inari Sami	Tetum
Indonesian	Thangmi
Interlingua	Tongan
Inuktitut (Latin)	Turkish
Irish	Turkmen (Latin)

Italian	Tuvan
Japanese	Upper Sorbian
Jaunsari (Devanagiri)	Urdu
Javanese	Uyghur (Arabic)
Kabuverdianu	Uzbek (Arabic)
Kachin (Latin)	Uzbek (Cyrillic)
Kangri (Devanagiri)	Uzbek (Latin)
Karachay-Balkar	Volapük
Kara-Kalpak (Cyrillic)	Walser
Kara-Kalpak (Latin)	Welsh
Kashubian	Western Frisian
Kazakh (Cyrillic)	Yucatec Maya
Kazakh (Latin)	Zhuang
Khaling	Zulu

Change History

Date of Change	Changed By	Summary of Change
March 2024	Martin Heikkilä	First version