

DataChecker.

DataChecker API Integration

Environments

We have two environments available for our customers, a staging environment for integration and testing and a production environment. For each environment you will receive different credentials.

Environment	API Base url
Staging	https://developer.staging.datachecker.nl
Production	https://developer.datachecker.nl

Authentication

The API is secured with an oAuth token, to retrieve a token you can submit a POST request to the `/api/v2/oauth/token` endpoint

This endpoint is secured with basic authentication, you will need the provided client ID and the secret to generate the basic authorization header.

The returned token can be used for multiple request as long as the token is not expired (see expiresIn return parameter)

Sample request

```
1 {
2   "scopes": [
3     "productapi.idverify.write",
4     "productapi.sdk.read",
5     "productapi.secureidlink.write"
6   ]
7 }
```

Sample response

HTTP Status: 200

```
1 {  
2   "accessToken":  
    "eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.eyJhdWQiOiJj0t1lNmViZC03NDZkLTRmYzQ0TUyYi1kZjEzZjEzNDY0ODI0OWIiLCJleHAiOjE3MTIwNjk0MDIsImRpZCI6ImVlbnJkZWZlYWlyNGltNDZmZC1hbmhLTQ4NWII2NWVwZGI0Sj9.VD3ceZjCXe9BkufMEAVZytw_Q52xdzy7NHpZGnq-zpFS7Q0GL9lNFTAiraIxMp2cnMK23RFr79zDAep8_UDDCg",  
3   "expiresIn": "3600",  
4   "tokenType": "Bearer"  
5 }
```

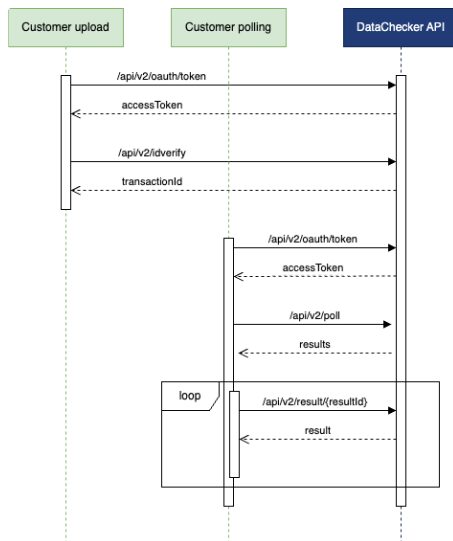
Sample error responses

HTTP Status: 401

```
1 {
2   "code": 16,
3   "message": "Unauthorized"
4 }
```

Workflow

It's recommended that you implement two separate processes. One to handle the upload of the documents and another process that will handle the processing of all the results. The following sequence diagram is an illustration of the processes.



API Calls

FaceVerify

The FaceVerify endpoint (`/api/v2/faceverify`) can be called with a POST request to upload the selfie challenges that are created by the FACE_VERIFY SDK.

Required scope: `productapi.faceverify.write`

Sample request

```

1 {
2   "transactionId": "{{TRANSACTION_ID}}",
3   "images": [
4     {
5       "type": "COMPARE",
6       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
7     },
8     {
9       "type": "LIVE",
10      "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
11     },
12     {
13       "type": "LIVE",
14       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
15     },
16     {
17       "type": "LIVE",
18       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
19     },
20     {
21       "type": "LIVE",
22       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
23     },
24     {
25       "type": "LIVE",
26       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFfcSAAAAADULEQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
27     }
28   ]
29 }
  
```

Field	Type	Optional	Description
transactionId	string	No	The transactionId that is retrieved when creating the SDK Token
images	array	No	A list of images. One compare image and multiple live images are required. The live images are created by the FACE_VERIFY SDK.

images/customerReference	string	Yes	A field that can be used to add a customer specific reference for this image
images/data	string	No	A base64 string of the image
images/type	string	No	The type of image. Possible values are COMPARE, LIVE

Sample response

HTTP Status: 200

```
1 {
2   "success": true
3 }
```

Sample error response

HTTP Status: 400

```
1 {
2   "code": 3,
3   "message": "NotEnoughLivePhotos"
4 }
```

IDVerify

The IDVerify endpoint (`/api/v2/idverify`) can be called with a POST request to upload the identity document for an IDV_LITE or IDV_STANDARD product. If you want to verify if the person is allowed to work, then you can upload the identity document together with a BSN Statement and/or a TWV document, as a product you can set RTW_STANDARD.

Required scope: `productapi.idverify.write`

Sample request

HTTP Status: 200

```
1 {
2   "product": "IDV_STANDARD",
3   "images": [
4     {
5       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
6     },
7     {
8       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
9     }
10  ],
11   "customerReference": "reference-id"
12 }
```

Field	Type	Optional	Description
product	string	No	Possible values are IDV_LITE, IDV_STANDARD or RTW_STANDARD
images	Array	No	A list of images
images/customerReference	string	Yes	A field that can be used to add a customer specific reference for this image
images/data	string	No	A base64 string of the image
images/type	string	Yes	The type of image. Possible values are ADDITIONAL, BSN_STATEMENT, COMPARE, DRIVING_LICENSE, IDENTITY_CARD, LIVE, PAPER, PASSPORT, PAYROLL_TAX_STATEMENT, PORTRAIT, RESIDENCE_PERMIT, RTW_DOCUMENT, SIGNATURE, WORK_PERMIT
customerReference	string	Yes	A field that can be used to add a customer specific reference for this transaction
userId	string	Yes	A valid userId for a user that is defined for your account

Additional documents

It is possible to add additional documents to the transactions by supplying the “ADDITIONAL” type to the image. The driving license check is integrated for the combination of the “ADDITIONAL” `type` and the “DRIVING_LICENSE” `additionalDocumentType` as long as this is enabled on your customer settings and the product is RTW_STANDARD.

```
1 {
2   "product": "RTW_STANDARD",
3   "images": [
4     {
5       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
6     },
7     {
8       "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg=="
9     },
10    {
11      "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
12      "type": "ADDITIONAL",
13      "additionalDocumentType": "DRIVING_LICENSE"
14    },
15    {
16      "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
17      "type": "ADDITIONAL",
18      "additionalDocumentType": "DRIVING_LICENSE"
19    }
20  ]
21 }
```

Sample response

HTTP Status: 200

```
1 {
2   "transactionId": "6a95ef92-bbbf-47a6-8c4b-cf3828cb3162"
3 }
```

Field	Type	Optional	Description
transactionId	string	No	The unique ID for the transaction, this ID is used in the polling and in the final result.

Sample error responses

HTTP Status: 400

```
1 {
2   "code": 3,
3   "message": "ImagesRequired"
4 }
```

HTTP Status: 400

```
1 {
2   "code": 3,
3   "message": "ImageDataRequired"
4 }
```

Poll

Once the documents are uploaded, you can start calling the polling endpoint (`/api/v2/poll`) with a GET request to see if a result is available. The preferred way to do this is to have a separate polling workflow that will check the results of all your transactions every 60 seconds. If you have not retrieved the polling record or the result within your data retention settings, then the data will be removed.

Required scope: `productapi.poll.read`

Sample response

HTTP Status: 200

```
1 {
2   "results": [
3     {
4       "resultId": "1317e999-1ba3-4699-9e60-035af976b76c",
```

```

5     "transactionId": "ae663ac6-60c1-4cc4-aa72-f05a7c91e68a",
6     "product": "IDV_PREMIUM",
7     "customerReference": "reference-id",
8     "completed": "2023-07-31T08:02:17.499Z"
9   },
10  {
11    "resultId": "7d4a4704-563e-4cec-8eef-ea98c5e4e870",
12    "transactionId": "268b8961-0e60-473e-ae03-6d90e32d6c64",
13    "product": "IDV_LITE",
14    "customerReference": "reference-id-other",
15    "completed": "2023-07-31T08:05:19.528Z"
16  }
17 ]
18 }

```

Field	Type	Optional	Description
resultId	string	No	The unique ID that can be used to retrieve the data from the result endpoint
transactionId	string	No	The unique ID of the transaction, this ID is also returned when the transaction was created
product	string	No	The product of the transaction
customerReference	string	Yes	The reference that was supplied when the transaction was created
completed	datetime	No	The UTC timestamp on which the transaction was completed
userId	string	Yes	The userId which is used to request the transaction

Result Webhook

Instead of polling for the results we can also configure a result webhook. This will send a POST request with the metadata to a url when the result is ready.

Sample request

```

1 {
2   "resultId": "1317e999-1ba3-4699-9e60-035af976b76c",
3   "transactionId": "ae663ac6-60c1-4cc4-aa72-f05a7c91e68a",
4   "product": "IDV_PREMIUM",
5   "customerReference": "reference-id",
6   "completed": "2023-07-31T08:02:17.499Z"
7 }

```

Result

You can use the resultId of the polling or webhook result to retrieve the data by calling the `/api/v2/result/{resultId}` endpoint with a GET request. When you have successfully retrieved the result, the the corresponding record in the polling result will be removed. You will still be able to retrieve the result with the provided resultId as long as the data is available (depending on your data retention settings).

Required scope: `productapi.result.read`

Sample response (product: RTW_PREMIUM)

HTTP Status: 200

```

1 {
2   "result": "APPROVED",
3   "transactionId": "ed04064c-1165-4afb-b2b1-96b22f6f1552",
4   "faceVerify": {
5     "result": "APPROVED"
6   },
7   "identity": {
8     "result": "APPROVED",
9     "data": {
10      "name": {
11        "fullName": "WILLEKE LISELOTTE DE BRUIJN",
12        "givenNames": "WILLEKE LISELOTTE",
13        "surname": "DE BRUIJN"
14      },
15      "dateOfBirth": "1965-10-03",

```

```
16     "dateOfIssuance": "2014-09-03",
17     "dateOfExpiry": "2024-09-03",
18     "documentType": "PASSPORT",
19     "documentNumber": "SPECI2014",
20     "gender": "F",
21     "issuingStateOrOrganization": "NLD",
22     "nationality": "NLD",
23     "placeOfBirth": "AMSTERDAM",
24     "personalGovernmentId": "999999990",
25     "residencePermitType": "I"
26 },
27 "documentValidation": {
28     "backgroundPrint": "APPROVED",
29     "font": "APPROVED",
30     "hologram": "APPROVED",
31     "lamination": "APPROVED",
32     "mrzFont": "APPROVED",
33     "opticallyVariableInk": "NOT_APPLICABLE",
34     "perforation": "NOT_APPLICABLE",
35     "photoIntegration": "APPROVED",
36     "stamps": "APPROVED"
37 },
38 "verifications": {
39     "lostAndStolen": {
40         "messages": [
41             "Document has been reported stolen"
42         ],
43         "verificationResult": "HIT"
44     },
45     "sanctionList": {
46         "foundOnLists": [
47             "CANADA",
48             "EUROPEAN_UNION",
49             "OFAC",
50             "RIJKSOVERHEID",
51             "UNITED_NATIONS"
52         ],
53         "verificationResult": "HIT"
54     }
55 },
56 },
57 "rightToWork": {
58     "result": "APPROVED",
59     "data": {
60         "dueDate": "2024-07-01",
61         "personalGovernmentId": "999999990",
62         "startDate": "2024-03-01",
63         "twvDueDate": "2024-12-31",
64         "twvFromDate": "2024-01-01",
65         "vsDueDate": "2024-07-01",
66         "vsFromDate": "2024-03-01"
67     }
68 },
69 "vog": {
70     "createdAt": "2025-01-01T12:00:00.000Z",
71     "status": "OPEN"
72 },
73 "images": [
74     {
75         "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
76         "documentType": "PASSPORT",
77         "pageType": "FRONT"
78     },
79     {
80         "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
81         "documentType": "PASSPORT",
82         "pageType": "BACK"
83     },
84     {
85         "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
86         "documentType": "BSN_STATEMENT",
87         "pageType": "ANY"
88     },
89     {
90         "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
91         "documentType": "WORK_PERMIT",
92         "pageType": "ANY"
93     },
94     {
95         "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAAFfcSjAAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAAABJRU5ErkJggg==",
96         "documentType": "COMPARE"
97     },
98     {
```

```

99     "data": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAFcSjAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRU5ErkJggg==",
100     "documentType": "LIVE"
101   }
102 ]
103 }
104

```

Field	Type	Optional	Description
result	string	No	The overall result of the transaction, possible values are APPROVED, DISAPPROVED, TECHNICAL_REJECT
messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
transactionId	string	No	The unique ID of the transaction
customerReference	string	Yes	The reference that was supplied when the transaction was created
drivingLicense	object	Yes	Only included if the driving license check is applied
drivingLicense/result	string	No	The result of the driving license check
drivingLicense/messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
drivingLicense/customerReferences	array of strings	Yes	A list of customer references that refer to the images that are used for this result object
drivingLicense/data	object	Yes	The data that is extracted from the driving license this can contain these fields: • name (object) • dateOfBirth (date, "yyyy-MM-dd") • dateOfIssuance (date, "yyyy-MM-dd") • dateOfExpiry (date, "yyyy-MM-dd") • documentNumber (string) • issuingStateOrOrganization (string) • personalGovernmentId (string) • class (string) • classes (array of classes containing the name and dateOfExpiry of each class)
drivingLicense/data/name	object	Yes	The name object that contains the name fields: • fullName (string) • givenNames (string) • surname (string)
faceVerify	object	Yes	Only included for the transactions started with the faceverify endpoint or for the premium products
faceVerify/result	string	No	The result of the face verification
faceVerify/messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
faceVerify/customerReferences	array of strings	Yes	A list of customer references that refer to the images that are used for this result object
identity	object	Yes	Included in all the calls to the idverify endpoint
identity/result	string	No	The result of the identity verification

identity/messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
identity/customerReferences	array of strings	Yes	A list of customer references that refer to the images that are used for this result object
identity/data	object	Yes	The data that is extracted from the identification document, this can contain these fields: • name (object) • dateOfBirth (date, "yyyy-MM-dd") • dateOfIssuance (date, "yyyy-MM-dd") • dateOfExpiry (date, "yyyy-MM-dd") • documentType (string) • documentSubType (string) • documentNumber (string) • gender (string) • issuingStateOrOrganization (string) • nationality (string) • personalGovernmentId (string) • residencePermitType (string)
identity/data/name	object	Yes	The name object that contains the name fields: • fullname (string) • givenNames (string) • surname (string)
identity/documentValidation	object	Yes	The checked security features of the identification document, each field can contain these values: APPROVED, NOT_APPLICABLE. The object can contain these fields: • backgroundPrint (string) • font (string) • hologram (string) • lamination (string) • mrzFont (string) • opticallyVariableInk (string) • perforation (string) • photoIntegration (string) • stamps (string)
identity/verifications	object	Yes	The result of additional verifications, e.g. the check against sanction lists
identity/verifications/lostAndStolen	object	Yes	The lost and stolen check
identity/verifications/lostAndStolen/messages	array of strings	Yes	If the result of this check is HIT, this field will list the reason(s) why
identity/verifications/lostAndStolen/verificationResult	string	Yes	The result of the lost and stolen check. Can have the following values: HIT, NO_HIT
identity/verifications/sanctionList	object	Yes	The sanction list check
identity/verifications/sanctionList/verificationResult	string	Yes	The result of the sanction list check. Can have the following values: HIT, NO_HIT

identity/verifications/sanctionList/foundOnLists	array of strings	Yes	If the value of the field above is HIT, this will list the sanction lists on which the person was found
rightToWork	object	Yes	This is only included if a RTW product is requested
rightToWork/result	string	No	The result of the right to work verification
rightToWork/messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
rightToWork/customerReferences	array of strings	Yes	A list of customer references that refer to the images that are used for this result object
rightToWork/data	object	Yes	The data that is extracted from the right to work documents. The object can contain these fields: For RTW Netherlands: • vsFromDate (date, "yyyy-MM-dd") • vsDueDate (date, "yyyy-MM-dd") • twvFromDate (date, "yyyy-MM-dd") • twvDueDate (date, "yyyy-MM-dd") • personalGovernmentId (string) For RTW Belgium: • attachmentDueDate (date, "yyyy-MM-dd") Generic: • dueDate (date, "yyyy-MM-dd") • startDate (date, "yyyy-MM-dd")
payrollTaxStatement	object	Yes	This is only included if a single payroll tax statement has been uploaded
payrollTaxStatement/result	string	No	The result of the payroll tax statement verification
payrollTaxStatement/messages	array of strings	Yes	A list of messages when the result is DISAPPROVED or TECHNICAL_REJECT
payrollTaxStatement/data	object	Yes	This is only included if a single payroll tax statement has been uploaded. This can contain these fields: • personalGovernmentIdPresent (boolean) • signDatePresent (boolean) • nameAddressResidencePresent (boolean) • signaturePresent (boolean) • applyPayrollTax (boolean) • payrollTaxDate (datetime) • signDate (datetime) • personalGovernmentId (string)
vog	object	Yes	This section is only included if a VOG is requested
vog/createdAt	string	No	The timestamp when the VOG was requested
vog/status	string	No	The status at the time of the VOG request
images	array	Yes	A list of images
images/customerReference	string	Yes	The reference for the image that was supplied when the transaction was created
images/data	string	No	A base64 string of the image

images/type	string	No	The type of image. Possible values are BSN_STATEMENT, COMPARE, DRIVING_LICENSE, IDENTITY_CARD, LIVE, PAPER, PASSPORT, PORTRAIT, RESIDENCE_PERMIT, RTW_DOCUMENT, SIGNATURE, WORK_PERMIT
images/pageType	string	No	The page type of the image. Possible values are FRONT, BACK, RESIDENCE_PERMIT_STICKERS, ANY
files	array	Yes	A list of files
files/data	string	No	A base 64 string of the file
files/type	string	No	The type of the file. Possible value is TRANSACTION_REPORT or PAYROLL_TAX_STATEMENT
userId	string	Yes	The userId which is used to request the transaction

Sample response (product: RTW_STANDARD with driving license check)

HTTP Status: 200

```

1  {
2    "result": "APPROVED",
3    "transactionId": "ed04064c-1165-4afb-b2b1-96b22f6f1552",
4    "drivingLicense": {
5      "result": "APPROVED",
6      "data": {
7        "name": {
8          "fullName": "WILLEKE LISELOTTE DE BRUIJN",
9          "givenNames": "WILLEKE LISELOTTE",
10         "surname": "DE BRUIJN"
11       },
12       "dateOfBirth": "1965-10-03",
13       "dateOfIssuance": "2015-09-03",
14       "dateOfExpiry": "2025-09-03",
15       "issuingStateOrOrganization": "NLD",
16       "nationality": "NLD",
17       "personalGovernmentId": "999999990",
18       "class": "AM-B",
19       "classes": [
20         {
21           "name": "A",
22           "dateOfExpiry": "2025-09-03"
23         },
24         {
25           "name": "AM",
26           "dateOfExpiry": "2025-09-03"
27         },
28         {
29           "name": "B",
30           "dateOfExpiry": "2025-09-03"
31         }
32       ]
33     },
34   },
35   "identity": {
36     "result": "APPROVED",
37     "data": {
38       "name": {
39         "fullName": "WILLEKE LISELOTTE DE BRUIJN",
40         "givenNames": "WILLEKE LISELOTTE",
41         "surname": "DE BRUIJN"
42       },
43       "dateOfBirth": "1965-10-03",
44       "dateOfIssuance": "2014-09-03",
45       "dateOfExpiry": "2024-09-03",
46       "documentType": "PASSPORT",
47       "documentNumber": "SPECI2014",
48       "gender": "F",
49       "issuingStateOrOrganization": "NLD",
50       "nationality": "NLD",
51       "placeOfBirth": "AMSTERDAM",
52       "personalGovernmentId": "999999990",
53       "residencePermitType": "I"
54     },
55     "documentValidation": {

```

```

56     "backgroundPrint": "APPROVED",
57     "font": "APPROVED",
58     "hologram": "APPROVED",
59     "lamination": "APPROVED",
60     "mizFont": "APPROVED",
61     "opticallyVariableInk": "NOT_APPLICABLE",
62     "perforation": "NOT_APPLICABLE",
63     "photoIntegration": "APPROVED",
64     "stamps": "APPROVED"
65 },
66 "verifications": {
67     "lostAndStolen": {
68         "messages": [
69             "Document has been reported stolen"
70         ],
71         "verificationResult": "HIT"
72     },
73     "sanctionList": {
74         "foundOnLists": [
75             "CANADA",
76             "EUROPEAN_UNION",
77             "OFAC",
78             "RIJKSOVERHEID",
79             "UNITED_NATIONS"
80         ],
81         "verificationResult": "HIT"
82     }
83 }
84 },
85 "rightToWork": {
86     "result": "APPROVED",
87     "data": {
88         "dueDate": "2024-07-01",
89         "personalGovernmentId": "999999990",
90         "startDate": "2024-03-01"
91         "twvDueDate": "2024-12-31",
92         "twvFromDate": "2024-01-01",
93         "vsDueDate": "2024-07-01",
94         "vsFromDate": "2024-03-01"
95     }
96 },
97 "vog": {
98     "createdAt": "2025-01-01T12:00:00.000Z",
99     "status": "OPEN"
100 },
101 "images": [
102     {
103         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
104         "documentType": "PASSPORT",
105         "pageType": "FRONT"
106     },
107     {
108         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
109         "documentType": "PASSPORT",
110         "pageType": "BACK"
111     },
112     {
113         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
114         "documentType": "ADDITIONAL",
115         "additionalDocumentType": "DRIVING_LICENSE"
116     },
117     {
118         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
119         "documentType": "ADDITIONAL",
120         "additionalDocumentType": "DRIVING_LICENSE"
121     }
122     {
123         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
124         "documentType": "BSN_STATEMENT",
125         "pageType": "ANY"
126     },
127     {
128         "data": "iVBORw0KGgoAAAANSUgAAAAEAAAABCAYAAAAFfcSJAAADU1EQVR42mP8/5+hHgAHggJ/PchI7wAAAABJRUSErkJggg==",
129         "documentType": "WORK_PERMIT",
130         "pageType": "ANY"
131     }
132 ]
133 }
134

```

Sample error response

HTTP Status: 404

```
1 {
2   "code": 5,
3   "message": "Result not found for ID: 9ab3e3a3-0316-4374-9561-54919d09c021"
4 }
```

SDK Token

To be able to use one of our SDK's, you will need to request a SDK Token to unlock the SDK. A SDK Token can be retrieved by sending a GET request to the `/api/v2/sdk/token` endpoint. Every transaction requires a new SDK Token.

Required scope: `productapi.sdk.read`

Sample request

```
/api/v2/sdk/token?
numberOfChallenges=3&customerReference=SomeReference&validateWatermark=true&services=FACE_VERIFY&services=AUTO_CAPTURE
```

Field	Type	Optional	Description
numberOfChallenges	int	Yes	The amount of challenges that need to be performed for FACE_VERIFY. The default is 2.
validateWatermark	boolean	Tes	When this is enabled, then an invisible watermark will be added to the images taken by the FACE_VERIFY SDK. This watermark will be checked when the images are uploaded to the faceverify endpoint.
services	Array of string	No	The name of the SDK that you want to unlock. Possible values are AUTO_CAPTURE, FACE_VERIFY or NFC
customerReference	string	Yes	A field that can be used to add a customer specific reference for this transaction.

Sample response

HTTP Status: 200

```
1 {
2   "token":
3     "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJjbGxvd2VkaXNka19saXN0IjpuZDVsL0JjaGZsb6VuZ2VzIjpbInJpZ2h0Iiwib6VmdCIsInVwI10sInRyYW5zYW50aW9uX21kIjoizM0YMDQwZmYtZjY4Mi00YTQxLWEzM2U0N2E1Mjc4NmM3ZjFiIiwiaXhwIjoxNzEyMTM4NzI3fQ.jWJj7-2wJTWV4c0xLPF41obp6u8-PTEA8hkj8X_1e0IbB3tUQyg18BRCGCuoVSNd0GpfQF8Jy_eT-_JXtQIwHD9g4tyTWEyaibakvn64EpKrJHF5UqCzU-VajacaLsmD2KNdQ0s_6fhbr-1a-YlY8W74LNOAjMLLgjpjB9wRwSbJrt1Uy91Sv5vPfjWLR8BoJZ1cigT1Mr806ihhQ10XczQqJpyH00BzpMBK5JwA4s0je1oNYnngMxKkvzfaozGIVW-ME4y5Qv7jLvzFCABeB8nnjaLw44iFGeg0tNo3T9zyj0A-j73gZfw1Ngk9XR192wIJ5UxePw69QSF1WAKrhzwkzAX3hrd-5_FWgB1HXiU1AVoQp998s7-cqAq3HNqYRGAnx6bv6Wh0j7Xi56WIZhVBqKBfMnw68MEJ0jTGRY116zn0o_x0G76YyApnZpdBuned2nTSxsNUTznhZTsoK1gk-7Ssz2aio1zzoKK0VAmjmfKXUubc0C88Q2PX4eje642FzQsfu0C0sqa1BmsM2zQ6ALG-12nuFJ8Xp4m64I1F91HZpUZm2yjt-zRydmQKaK6R0isgnEbALQtuwsU7SC00mq6rYD4eUc61xcvIqkJSFEdyxXLSgFsVcvbU6d6zkbrFYN_euw6UeW7t6CmP55jpBQM DYCA9KN_Y3c",
4   "transactionId": "fe2040ff-f682-4a41-a33e-7a52786c7f1b"
```

Field	Type	Optional	Description
token	string	No	The token
transactionId	string	No	The unique ID for the transaction, this ID is used in the polling and in the final result.

Sample error response

HTTP Status: 400

```
1 {
2   "code": 3,
3   "message": "parsing list \"services\": \"UNKNOWN\" is not a valid value"
4 }
```

SecureIDLink

The `/api/v2/secureidlink` endpoint can be used to create a Secure ID Link.

Required scope: `productapi.secureidlink.write`

Sample request

```
1 {
2   "product": "IDV_PREMIUM",
3   "emailAddress": "willeke.de.bruijn@datachecker.nl",
4   "communicationModes": [
5     "MODE_EMAIL",
6     "MODE_SMS"
7   ],
8   "disableCommunication": false,
9   "enableOneTimePassword": true,
10  "languagePrefix": "nl",
11  "name": "Willeke de Bruijn",
12  "phoneNumber": "+31854012650",
13  "customerReference": "reference-id"
14 }
```

Field	Type	Optional	Description
product	string	No	Possible values are IDV_LITE, IDV_STANDARD, IDV_PREMIUM, RTW_STANDARD and RTW_PREMIUM
emailAddress	string	Yes	The email address of the user that will receive the Secure ID Link
communicationModes	Array of strings	No	Possible values are MODE_EMAIL or MODE_SMS.
disableCommunication	boolean	Yes	When this is set to true, then the link and QR code will be returned in the response. The configured email and sms will not be sent.
enableOneTimePassword	boolean	Yes	This can be used to enable the one time password when an end user starts the onboarding process.
languagePrefix	string	No	The initial language that will be used on the Secure ID Link and in the communication with the end user. This is the two letter country code in lowercase.
name	string	No	The full name of the end user, this will be used in the communication.
phoneNumber	string	Yes	The phone number if the MODE_SMS is used. This should be in the international format.
customerReference	string	Yes	A field that can be used to add a customer specific reference for this transaction.
userId	string	Yes	A valid userId for a user that is defined for your account

Additional documents

It is possible to add additional documents to the transactions by adding the `additionalDocuments` parameter. A special workflow is integrated for the "DRIVING_LICENSE" as long as this is enabled on your customer settings and the product is RTW_STANDARD or RTW_PREMIUM.

```
1 {
2   "product": "RTW_PREMIUM",
3   "emailAddress": "willeke.de.bruijn@datachecker.nl",
4   "communicationModes": [
5     "MODE_EMAIL"
6   ],
7   "disableCommunication": false,
8   "languagePrefix": "nl",
9   "name": "Willeke de Bruijn",
10  "customerReference": "Customer reference",
11  "additionalDocuments": [
```

```

12  "DRIVING_LICENSE"
13  ]
14  }

```

Sample response

HTTP Status: 200

```

1  {
2    "link": "https://secureidlink.staging.datachecker.nl/05be80c6-c105-4f34-bf62-beb0ef6da2a5",
3    "secureId": "05be80c6-c105-4f34-bf62-beb0ef6da2a5",
4    "transactionId": "d097dc8a-a892-470b-a3be-302a37e557d4",
5    "qrCode":
      "iVBORw0KGgoAAAANSUuEugAAAQAAAAEAQMAAABmVDoIAAAAB1BMVEX//8AAABVwtN+AAACPU1EQVR42uyZwY3jMAxFv66DjipBpagz28E0p1JUgo466P6LT3n6yWbvK+2GhyA
      w3gBDmPz8ZPCJT/yXsZDkCSDsgCNianFH0tM8EdAArEgIu88sCL1F1qS/mAQIrkVPrm4ksJENW02ex4AA60Ie+t/JHZMCdNx9djyRgKX0B6iifD70zZWN6X8qub8MWPP6fIRHS66
      sKvtHfevuwYEeSnNreg3MDeu7nA40qIKpPpIjk6eJgTHQAAAFU29r0I3hfVZwLUt6f61qXrS4WpPfn1XVFTAIi0agk+2CnNhIy4h6kAaJ51m7zX0DgDaWo1D0D0pKpBYF6ViGqo
      XAvdUUXMAq5pjNZ0jPrf4eRdzAMUgwxncmVTclsFy53mCACVgD4s1ZB9kVXr6CzAooSWpnFgTkeTYMdTRU0BoM8zSIdg3qdhDdSDPAxgfhJIh20h5tLiKeB1ARKdgc90rtiWwV5
      RE1J/+4c5gF5HTgvWLaSI+0CALU9bS30j1KTyssa3Ts4BWDfIqm2t8y3umryYC4hHN/Mb+2NZB/mHipEAW1YpJ2bNK2//XFGzAGreHz2YTIIfuc9MUgDYUz00G87rk2Ph9KrkZgGg
      uTWtKF88r5WMg4HvvNmt8C2kvq2mA68pqaxTtid1A/M+6MgfQ71HkdcB8T3MI4LrsVb2QooXqTUinAUxIadus+pa+vBj0iYCGfo/q6jMWYL8dA0G6AgNW9q8mZ3igX11ShIfpfui
      D6/fuHhz4xCf+sfgVAAD//9YXywjwx2jAAAAE1FTKSuQmCC"
6    "communicationsResponse": [
7      {
8        "communicationMode": "MODE_SMS",
9        "status": "SENT"
10     },
11     {
12       "communicationMode": "MODE_EMAIL",
13       "status": "SENT"
14     }
15   ]
16 }

```

Field	Type	Optional	Description
link	string	Yes	The link for the Secure ID Link, this will only be returned if the disableCommunication option is set to true.
secureId	string	No	The unique ID for the link
transactionId	string	No	The unique ID for the transaction, this ID is used in the polling and in the final result.
qrCode	string	Yes	A base64 encoded image of a QR code that contains the Secure ID Link. This will only be returned if the disableCommunication option is set to true.
communicationsResponse	Array of objects	No	An array of objects that will indicate if the message has been sent. The delivery of the message could still fail, this can take several minutes. Each object contains these fields: - communicationMode (possible values are: MODE_SMS or MODE_EMAIL) - status (possible values are: SENT or FAILED)

Sample error response

HTTP Status: 400

```

1  {
2    "code": 3,
3    "message": "Name cannot be empty"
4  }

```

SecureIDLink Upload Webhook

When the SecureIDLink is used to upload the documents, we can send a POST request with the metadata to a url. This webhook can be configured for your account.

Sample request

```

1  {
2    "transactionId": "ae663ac6-60c1-4cc4-aa72-f05a7c91e68a",
3    "product": "IDV-PREMIUM",

```

```

4  "customerReference": "reference-id",
5  "type": "UPLOAD"
6  }

```

VOG Request

The `/api/v2/vog` endpoint can be used to create a VOG request.

Required scope: `productapi.vog.write`

Sample request

```

1  {
2    "customerReference": "your reference",
3    "emailAddress": "willeke@datachecker.nl",
4    "givenNames": "Willeke",
5    "language": "nl",
6    "surname": "De Bruijn",
7    "userId": "",
8    "vogRequest": {
9      "companyName": "DataChecker",
10     "jobTitle": "Developer",
11     "profiles": [ "11", "12" ],
12     "profileType": "GENERIC"
13   }
14 }

```

Field	Type	Optional	Description
customerReference	string	Yes	A field that can be used to add a customer specific reference for this transaction.
emailAddress	string	No	The email address of the user that will receive the Secure ID Link
givenName	string	No	The given names of the person that will request the VOG
language	string	No	The initial language that will be used on the Secure ID Link and in the communication with the end user. This is the two letter country code in lowercase.
surname	string	No	The surname of the person that will request the VOG
userId	string	Yes	A valid userId for a user that is defined for your account
vogRequest	object	No	The object to request a VOG
vogRequest/companyName	string	No	A required field for the company name
vogRequest/jobTitle	string	No	A required field for the job title
vogRequest/profiles	Array of strings	No	A list of profile numbers belonging to the VOG profiles that need to be checked, see Screeningsprofielen voor werkgevers Justis
vogRequest/profileType	string	No	The VOG profile type. Possible values are GENERIC and SPECIFIC.

Mocked Responses

To test the different kind of results, you can add a “dc-test” http header to the request of the IDVerify and SecureIDLink endpoints. This will immediately provide a mocked result that will be available at the polling and result endpoint. This is only available on the staging environment.

The content of the mocked result can change when another product is requested.

- IDV products will contain the identity object

- RTW products will also contain the rightToWork object
- PREMIUM products will also contain the faceVerify object (only available on the SecureIDLink endpoint)

"dc-test" http header value	Result description
APPROVED	An approved result will be provided
APPROVED_EU	A Polish Passport will be returned
APPROVED_NON_EU	A Ukrainian Passport will be returned
APPROVED_RP	A Dutch Residence permit will be returned
DISAPPROVED	A disapproved result will be provided
TECHNICAL_REJECT	A rejected result will be provided