Berlin Group NextGenPSD2

Changelog

Argenta Bank- en Verzekeringsgroep nv, Belgiëlei 49-53, 2018 Antwerpen

PSD2 Compliance Berlin Group - Changelog 1.17.0

Introduction

The updated version published is following the international Berlin Group NextGenPSD2 standard, version 1.3. Please note that the version in the URL will not change. The "v1" will be kept in all existing URLs.

Initiating a payment without debtor account(Choose debtor account at Bank side)

With this feature, Payment and Periodic Payments can be initiated without debtor account information and the debtor account can be selected later by the user at ASPSP side.

This feature is configurable so bank can enable/disable it, by default its enabled. The example request structure of payment initiation without debtor can be found in the Implementation Guidelines

> **Note**: If explicit authorization has been chosen by using `tpp-explicit-authorisation-preferred` request header then It is mandatory to provide a debtor information in the request.

Addition of lastChangeDateTime in the balance details for the various AIS endpoints

As part of the performance improvement of the AIS flow, where in we can optimize the calls made by the TPP to fetch the transaction details of the account, the berlin group flag, `lastChangeDateTime` has been added in the balance field to the response of the endpoints, accounts with balance details, balances and transactions with balance details, to indicate if there is any change in the balance since the last call. This will allow the TPP to know if it is useful to perform additional call to transactions endpoint. The below mentioned endpoints are enhanced with this change:

- GET accounts?withBalance
- GET accounts/{account-id}?withBalance
- GET accounts/{account-id}/balances
- GET accounts/{account-id}/transactions?withBalance

Examples:

Request structure:

```
GET /berlingroup/v1/accounts/6022b7dc3e84d93908ab612a?withBalance=true
```

```
apiKey: {api-key}
digest: {digest}
signature: {signature}
authorization: {authorization}
Consent-ID: 62e7ceadd80f3b55c16a39e0
x-request-id: 39ea5893-c82a-4142-a452-c89f6c70da93
content-type: application/json;charset=UTF-8
tpp-signature-certificate: {certificate}
psu-ip-address: 192.168.1.135
psu-ip-port: 54321
```

Response structure:

```
HTTP/1.1 200 Ok
Date: Wed, 07 Jan 2019 03:07:30 GMT
request-id: b44fe34d-f9e3-4343-9ec7-e5f8100907b9
```


```
{
  "cashAccountType": "ODFT",
  "resourceId": "6022b7dc3e84d93908ab612a",
  "product": "Current account",
  "balances": [
    {
      "balanceType": "interimBooked",
      "balanceAmount": {
        "amount": "5.00",
        "currency": "EUR"
      },
      "referenceDate":"2019-02-07",
      "lastChangeDateTime": "2019-02-06T03:07:29.433Z"
    }
  ],
  "bban": "999000186794",
  "iban": "BE87999000186794",
  "name": "Mr Rohit Srivastava",
  "currency": "EUR",
  "_links": {
    "balances": {
      "href": "\/v1\/accounts\/6022b7dc3e84d93908ab612a\/balances"
    }
  }
}
```


Adaptation of Berlin Group field deltaList in GET transactions

DeltaList data attribute indicates that the AISP is in favour to get all transactions after the last report access for this PSU on the addressed account.

- When deltaList is false.
  - It is *mandatory* to send dateFrom parameter.
  - The flow to request transactions remains the same.
- When deltaList is true.
  - It is optional to send dateFrom parameter.
  - The dateTo will be taken as *today*.
  - The links formed contain the dateFrom and dateTo calculated appropriately according to the delta transactions to be provided. The links can then be used to get subsequent transactions.

Note: By default we would be considering the deltaList as *false* if not provided by the TPP explicitly.

Examples:

Request structure:

```
GET /berlingroup/v1/accounts/62b2befbe0e2807fd33b6cea/transactions?deltaList=true
```

```
apiKey: {api-key}
digest: {digest}
signature: {signature}
authorization: {authorization}
Consent-ID: 62e7ceadd80f3b55c16a39e0
x-request-id: 39ea5893-c82a-4142-a452-c89f6c70da93
content-type: application/json;charset=UTF-8
tpp-signature-certificate: {certificate}
psu-ip-address: 192.168.1.135
psu-ip-port: 54321
```

Response structure:

```
HTTP/1.1 200 Ok
Date: Mon, 08 Aug 2022 04:16:23 GMT
request-id: b44fe34d-f9e3-4343-9ec7-e5f8100907b9
```

```
{
  "_links": {
    "href": ""
  },
  "transactions": {
    "booked": [
      {
        "creditorName": "Dupont",
        "_links": {
          "href": ""
        },
```

```
      "transactionAmount": {
        "amount": "-10.50",
        "currency": "EUR"
      },
      "creditorAccount": {
        "iban": "BE81999000344624"
      },
      "proprietaryBankTransactionCode": "PGIPSTDB",
      "bookingDate": "2022-08-08",
      "valueDate": "2022-08-08",
      "remittanceInformationUnstructured": "toto ",
      "transactionDate": "2022-08-08",
      "entryReference": "C2H08PGEW63K1JDR"
    },
    {
      "creditorName": "Dupont",
      "_links": {
        "href": ""
      },
      "transactionAmount": {
        "amount": "-10.50",
        "currency": "EUR"
      },
      "creditorAccount": {
        "iban": "BE81999000344624"
      },
      "proprietaryBankTransactionCode": "PGCTSTDB",
      "bookingDate": "2022-08-08",
      "valueDate": "2022-08-08",
      "remittanceInformationUnstructured": "toto ",
      "transactionDate": "2022-08-08",
      "entryReference": "C2H08PGEUS7JNM3J"
    },
    {
      "creditorName": "Dupont",
      "_links": {
        "href": ""
      },
      "transactionAmount": {
        "amount": "-10.50",
        "currency": "EUR"
      },
      "creditorAccount": {
        "iban": "BE81999000344624"
      },
      "proprietaryBankTransactionCode": "PGCTSTDB",
      "bookingDate": "2022-08-08",
      "valueDate": "2022-08-08",
      "remittanceInformationUnstructured": "toto ",
      "transactionDate": "2022-08-08",
      "entryReference": "C2H08PGEUQ2JMZR9"
    }
  ],
  "_links": {
```

```
      "next": {
        "href": "/v1/accounts/62c2d325f201f37839eea82d/transactions?dateTo=2022-
08-17&bookingStatus=both&dateFrom=2022-05-19&range=3-5"
      },
      "last": {
        "href": "/v1/accounts/62c2d325f201f37839eea82d/transactions?dateTo=2022-
08-17&bookingStatus=both&dateFrom=2022-05-19&range=33-35"
      },
      "self": {
        "href": "/v1/accounts/62c2d325f201f37839eea82d/transactions?dateTo=2022-
08-17&bookingStatus=both&dateFrom=2022-05-19&range=0-2"
      },
      "first": {
        "href": "/v1/accounts/62c2d325f201f37839eea82d/transactions?dateTo=2022-
08-17&bookingStatus=both&dateFrom=2022-05-19&range=0-2"
      },
      "account": {
        "href": "/v1/accounts/62c2d325f201f37839eea82d"
      }
    }
  }
}
```


Adapted a proper error mapping for when account mandate is revoked at the bank backend

When mandate revoked for some consented accounts, and the user tries to fetch the details for these accounts, error was not according to BG documentation, now we have correct error code `RESOURCE_EXPIRED` for such incident.