Account Information API

MENU

1. API Flows

1.1 Step 1: Account Access Consent Request

The flow MUST begin with a PSU who provides consent for a TPP to access their account information from a PASP that they already have a relationship.

1.2 Step 2: Setup Account Access Consent

The TPP MUST request PSU access consent from the PASP. To support this, the TPP MUST:

- Successfully authenticate with the PASP authorization server to obtain an access token.
- Use the access token to create an access consent request against the PASP resource server.

The PASP MUST immediately acknowledge the PSU consent request from the TPP. The PASP MUST respond with an identifier for the access consent request (the ConsentId - which is the intent identifier).

This step MUST be fulfilled by a POST request to the /account-access-consents resource made available on the PASP resource server.

The account access consents resource request SHOULD include all REQUIRED data permissions that the PSU intends to provide to the TPP.

A TPP MAY be a broker for data to other parties, so it is valid for a PSU to have multiple consents for the same resource, with different consent or authorization parameters agreed upon.

1.2.1 Security and Access Control

Client Credentials Grant

- The TPP MUST obtain an access token (with the appropriate FAPI Security Profile scope) from the PASP authorization server using the Client Credentials grant type over a mutual TLS connection.

1.2.2 Subscribing to Events

Where both a TPP and PASP support event notifications (e.g., Webhooks), a TPP MAY subscribe their preferences with the PASP during the consent request.

Webhooks

- The TPP MUST register an HTTPS callback URL with the PASP to receive events.
- The PASP MUST publish events to the HTTPS callback URL specified by the TPP.
- The PASP MUST publish a single resource event to the TPP (e.g., An individual ConsentId or AccountID)
- The PASP MAY publish a single resource collection of events to the TPP for the same Event Type (e.g., An array of all transactions that have been created)

1.3 Step 3: Authorize Account Access Consent

The TPP MUST now request the PSU to authorize the consent. Please refer to the API User Guide, Section 5 Authorization Design, to review the supported Authorization Flows.

1.3.1 Security and Access Control

Client Credentials Grant

The TPP SHOULD use the Client Credentials grant to obtain a URI from the PASP authorization server that will support either the Redirect or Decoupled flows.

Authorization Code Grant

The TPP MUST use an authorization Code grant to obtain a token to access all other API resources.

1.4 Step 4: Access Data

1.4.1 Request Data

The TPP MUST have a valid access token (with scope) from the PASP authorization server.

The TPP MUST use the valid access token to retrieve PSU data from the PASP resource server.

The PASP MUST return an API response when provided with a valid access token request from the TPP.

1.4.2 Receive Data

The PASP MUST observe a state change event on resource data associated with the ConsentId and data permissions.

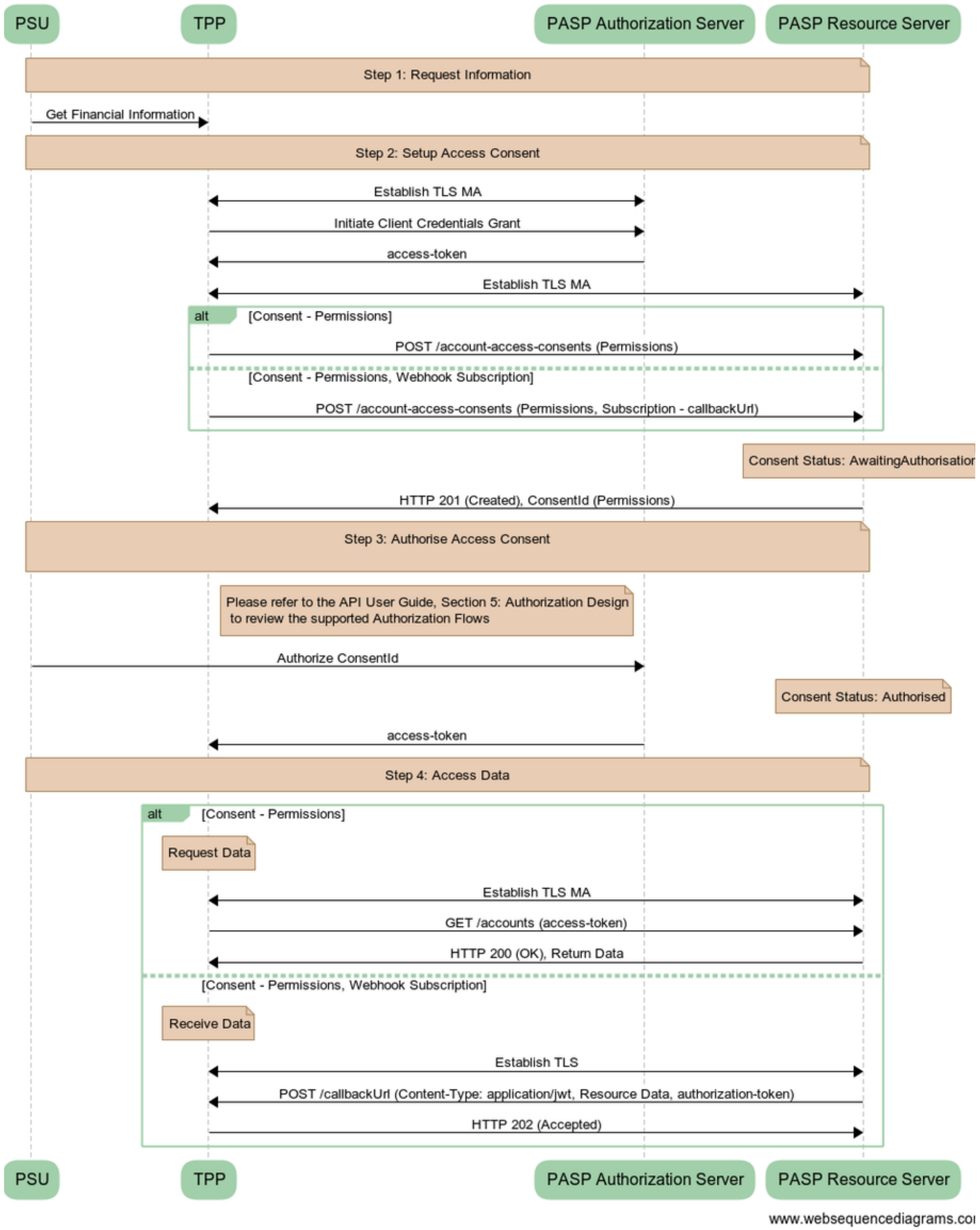
The PASP MUST generate a Self Signed JWT Authorization Token to provide to the TPP in the Authorization Header.

The PASP MUST publish the resource data to the TPP's callback URL (registered during the consent request)

The TPP MUST validate the Self Signed JWT Authorization Token from the PASP before storing the data and returning an HTTP status response to the PASP.

2. Sequence Diagram

The above Steps (1 to 4) are captured as a series of end-to-end flows with the relevant Actors. The flows illustrate the API interactions completing successfully, with no API Errors.


PSU
TPP
PASP Authorization Server
PASP Resource Server
Step 1: Request Information
Get Financial Information
Step 2: Setup Access Consent
Establish TLS MA
Initiate Client Credentials Grant
access-token
Establish TLS MA
alt
[Consent - Permissions]
POST /account-access-consents (Permissions)
[Consent - Permissions, Webhook Subscription]
POST /account-access-consents (Permissions, Subscription - callbackUrl)
Consent Status: AwaitingAuthorisatior
HTTP 201 (Created), ConsentId (Permissions)
Step 3: Authorise Access Consent
Please refer to the API User Guide, Section 5: Authorization Design to review the supported Authorization Flows
Authorize ConsentId
Consent Status: Authorised
access-token
Step 4: Access Data
alt
[Consent - Permissions]
Request Data
Establish TLS MA
GET /accounts (access-token)
HTTP 200 (OK), Return Data
[Consent - Permissions, Webhook Subscription]
Receive Data
Establish TLS
POST /callbackUrl (Content-Type: application/jwt, Resource Data, authorization-token)
HTTP 202 (Accepted)
PSU
TPP
PASP Authorization Server
PASP Resource Server
www.websequencediagrams.coı

3. Examples

The following are non-normative examples of API access and usage of the Account Information API.

3.1 The TPP Obtains an Access Token from the PASP

3.1.1 Request: The TPP Uses the Client Credentials Grant Type with Scope(s)

The following example illustrates the Authorization Server supporting use of the private_key_jwt Authorization method.

```
POST /token HTTP/1.1
Host: as1.lab.openbanking.sa
Accept: application/json
Content-Type: application/x-www-form-urlencoded
grant_type=client_credentials
&scope=accounts%20openid
&client_assertion=eyJraWQiOiJpcFFmOEowanhNWmtUWlpjMWlVaXNqWjNTaUdzQk0yVGFhOHJKdWdKYndrIiwiYWxnIjoiUFMyNTYifQ.eyJz
&client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-bearer
```

3.1.2 Response: The PASP Returns an Access Token to the TPP

```
HTTP/1.1 200 OK
{
    "access_token": "ad297304-1057-4c68-9e76-a96f300a27f1",
    "token_type": "Bearer",
    "expires_in": 600
}
```

3.2 The TPP Creates an Account Consent Request to the PASP for Permission to Access Their PSUs' Account Data

3.2.1 Request: Data Permissions

```
POST /open-banking/account-information/2022.11.01-final-errata2/account-access-consents HTTP/1.1
Host: rs1.lab.api.openbanking.sa
Accept: application/json
Content-Type: application/json
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Authorization: Bearer ad297304-1057-4c68-9e76-a96f300a27f1
{
    "Data": {
        "Permissions": [
            "ReadAccountsBasic",
            "ReadAccountsDetail",
            "ReadBalances",
            "ReadPAN",
            "ReadBeneficiariesBasic",
            "ReadBeneficiariesDetail",
            "ReadScheduledPaymentsBasic",
            "ReadScheduledPaymentsDetail",
            "ReadParty",
            "ReadPartyPSU",
            "ReadPartyPSUIdentity",
            "ReadTransactionsBasic",
            "ReadTransactionsCredits",
            "ReadTransactionsDebits",
            "ReadTransactionsDetail"
            ],
        "ExpirationDateTime": "2022-01-10T10:40:00+02:00",
        "TransactionFromDateTime": "2022-01-01T10:40:00+02:00",
        "TransactionToDateTime": "2022-12-31T10:40:00+02:00"
    }
}
```

3.2.2 Response: PASP Accepts the TPP Account Consent Request and Returns a ConsentId and Status

```
HTTP/1.1 201 Created
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Content-Type: application/json
{
  "Data": {
    "Permissions": [
      "ReadAccountsBasic",
      "ReadAccountsDetail",
      "ReadBalances",
      "ReadPAN",
      "ReadBeneficiariesBasic",
      "ReadBeneficiariesDetail",
      "ReadScheduledPaymentsBasic",
      "ReadScheduledPaymentsDetail",
      "ReadParty",
      "ReadPartyPSU",
      "ReadPartyPSUIdentity",
      "ReadTransactionsBasic",
      "ReadTransactionsCredits",
      "ReadTransactionsDebits",
      "ReadTransactionsDetail"
    ],
    "TransactionFromDateTime": "2022-01-01T10:40:00+02:00",
    "TransactionToDateTime": "2022-12-31T10:40:00+02:00",
    "ConsentId": "aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd",
    "Status": "AwaitingAuthorization",
    "CreationDateTime": "2022-08-07T08:01:41.344Z",
    "StatusUpdateDateTime": "2022-08-07T08:01:41.344Z"
  },
  "Links": {
    "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/accour
  },
  "Meta": {}
}
```

3.3 The TPP Redirects the PSU to Their PASP for Them to Consent and Approve Permissions to Access Their Account(s)

3.3.1 Request: TPP Uses PAR (Pushed Authorization Request) to Obtain a Redirection URI to the PASP

Initialize the PAR Request JWT. The scope contains the parameterized ConsentId.

```
{
  "typ": "JWT",
  "alg": "PS256",
  "kid": "e4ce77c498e77000a25aa7b40e4a83f9"
}
.
{
    "iss": "s6BhdRkqt3",
    "iat": 1669393154,
    "exp": 1669393496,
    "nbf": "1669393154",
    "aud": "https://server.example.com",
    "response_type": "code id_token",
    "redirect_uri": "https://pasp.lab.openbanking.sa/auth",
    "scope": "openid accounts:aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd",
    "state": "af0ifjsldkj"
}
```

Create the PAR Request using the signed and/or encrypted JWT. The request parameter JWT includes the ConsentId that was originally generated by the PASP in the Consent response. The `ConsentId` is provided as a parameterized scope.

```
POST /open-banking/v1/par HTTP/1.1
Host: auth1.lab.openbanking.sa
Content-Type: application/x-www-form-urlencoded
Accept: application/json
client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-bearer
&client_assertion=eyJhbGciOiJIUzI1NiJ9.ew0KICAiaXNzIjogImM4NDIyNzg3LTFkZmYtNDI0ZC1iNjIwLTM1NmMwODcwYmVkNCIsDQogIC
&request=eyJ0eXAiOiJKV1QiLCJhbGciOiJQUzI1NiIsImtpZCI6ImU0Y2U3N2M0OThlNzcwMDBhMjVhYTdiNDBlNGE4M2Y5In0.eyJpc3MiOiJz
```

3.3.2 Response: The PASP Provides the Redirection URI for the TPP to Follow

```
HTTP/1.1 201 Created
Content-Type: application/json
Cache-Control: no-cache, no-store
{
  "request_uri": "urn:ietf:params:oauth:request_uri:6esc_11ACC5bwc014ltc14eY22c",
  "expires_in": 60
}
```

3.4 The TPP Redirects the PSU to Their PASP via the URI so that PSU Consent with Their PASP Can Be Completed

```
GET /auth?client_id=c8422787-1dff-424d-b620-356c0870bed4&request_uri=urn:ietf:params:oauth:request_uri:6esc_11ACC
Host: PASP.lab.openbanking.sa
```

3.5 The PSU Logs into Their PASP, Reviews the Consent Request, and Confirms the Accounts They Want to Share with the TPP.

This API request example is not part of the Open Banking API Specification. It is an example only to show how a PASP can receive a confirmed account consent request from their PSU. It is up to each PASP to determine how they will capture and store account consents in their enterprise systems.

```
POST /auth/aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd/aac/doConfirm
host: auth1.lab.openbanking.sa
Content-Type: application/x-www-form-urlencoded
accounts=f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115
&accounts=bed6cb83-956e-4795-86c3-0f4254ae1cab
&accounts=528b9f0c-c4e1-45fd-8f28-ab53fda4c850
&accounts=fe1e15fe-d4aa-4b4c-9ce0-e69bbf901fa6
&accounts=802d03c3-4ac5-4809-8c1e-f9f046e314e4
&accounts=02d19fb7-cf51-4b9a-a958-77701120da3c
```

3.6 The PASP Returns an Authorization Code to the TPP

```
302 Found
Location: https://tpp1.lab.openbanking.sa/simple-redirect-url?
code=ce2aeabf-599c-4475-9171-1f6d8c1a49dc
&state=2616df22-899e-468b-b7af-927145b067cc
```

3.7 The TPP Exchanges the Authorization Code for an Account API Access Token

```
POST /token HTTP/1.1
Host: as1.lab.openbanking.sa
Content-Type: application/x-www-form-urlencoded
Accept: application/json
grant_type=authorization_code
&code=ce2aeabf-599c-4475-9171-1f6d8c1a49dc
&client_assertion_type=urn%3Aietf%3Aparams%3Aoauth%3Aclient-assertion-type%3Ajwt-bearer
&client_assertion=eyJhbGciOiJIUzI1NiJ9.ew0KICAiaXNzIjogImM4NDIyNzg3LTFkZmYtNDI0ZC1iNjIwLTM1NmMwODcwYmVkNCIsDQogIC
&redirect_uri=https%3A%2F%2Fauth1.lab.openbanking.sa%2Fsimple-redirect-url
```

3.8 The PASP Returns an Access Token, Refresh Token, and ID Token to the TPP

```
HTTP/1.1 200 OK
Content-Type:application/json
{
    "access_token": "caa1b60d-61ff-4cd8-a4e1-2d18c8696de0",
    "expires_in": 432000,
    "token_type": "Bearer",
    "scope": "openid%20accounts%3Aaac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd",
    "state": "2616df22-899e-468b-b7af-927145b067cc",
    "refresh_token": "266f5f15-eb81-4a02-bf05-e25063ca445f",
    "id_token": "eyJhbGciOiJQUzI1NiIsImtpZCI6IkM4a3FRRlZoUFVOUnZTN1ljamZBSEVSTEVDZEFfamZENXJjb1NXVkMwY2sifQ.eyJz
}
```

The TPP can now request account resources using the access token.

3.9 Get a List of Accounts

3.9.1 Request: Accounts Resource

```
GET /open-banking/account-information/2022.11.01-final-errata2/accounts HTTP/1.1
Host: rs1.lab.api.openbanking.sa
Accept: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
Authorization: Bearer caa1b60d-61ff-4cd8-a4e1-2d18c8696de0
```

3.9.2 Response: Accounts Resource

```
HTTP/1.1 200 OK
Content-Type: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
{
  "Data": {
    "Account": [
      {
        "AccountId": "f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115",
        "Currency": "SAR",
        "Account": [
          {
            "Name": "Account 1",
            "SchemeName": "KSAOB.SortCodeAccountNumber",
            "Identification": "00003130000001"
          }
        ]
      },
      {
        "AccountId": "bed6cb83-956e-4795-86c3-0f4254ae1cab",
        "Currency": "SAR",
        "Account": [
          {
            "Name": "Account 2",
            "SchemeName": "KSAOB.SortCodeAccountNumber",
            "Identification": "00003130000002"
          }
        ]
      },
      {
        "AccountId": "528b9f0c-c4e1-45fd-8f28-ab53fda4c850",
        "Currency": "SAR",
        "Account": [
          {
            "Name": "Account 3",
            "SchemeName": "KSAOB.SortCodeAccountNumber",
            "Identification": "00003130000003"
          }
        ]
      }
    ]
  },
  "Links": {
    "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/accour
  },
  "Meta": {
    "TotalPages": 1
  }
}
```

3.10 Get Balances for an Account

3.10.1 Request: accounts/{`AccountId`}/balances resource

```
GET /open-banking/account-information/2022.11.01-final-errata2/accounts/f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115/bala
Host: rs1.lab.api.openbanking.sa
Accept: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
Authorization: Bearer caa1b60d-61ff-4cd8-a4e1-2d18c8696de0
```

3.10.2 Response: accounts/{`AccountId`}/balances resource

```
HTTP/1.1 200 OK
Content-Type: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
{
    "Data": {
        "AccountId": "f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115",
        "Balance": [
            {
                "Amount": {
                    "Amount": "1500.00",
                    "Currency": "SAR"
                },
                "CreditDebitIndicator": "Credit",
                "Type": "ClosingAvailable",
                "DateTime": "2016-03-19T00:00:00.000Z",
                "CreditLine": [
                    {
                        "Included": true,
                        "Type": "Temporary",
                        "Amount": {
                            "Amount": "500.00",
                            "Currency": "SAR"
                        }
                    }
                ]
            },
            {
                "Amount": {
                    "Amount": "1500.00",
                    "Currency": "SAR"
                },
                "CreditDebitIndicator": "Credit",
                "Type": "InterimAvailable",
                "DateTime": "2016-03-19T00:00:00.000Z",
                "CreditLine": [
                    {
                        "Included": true,
                        "Type": "Temporary",
                        "Amount": {
                            "Amount": "500.00",
                            "Currency": "SAR"
                        }
                    }
                ]
            }
        ]
    },
    "Links": {
        "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/ac
    },
    "Meta": {
        "TotalPages": 1
    }
}
```

3.11 Get Transactions for an Account

3.11.1 Request: accounts/{`AccountId`}/transactions resource

```
GET /open-banking/account-information/2022.11.01-final-errata2/accounts/f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115/trar
Host: rs1.lab.api.openbanking.sa
Accept: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
Authorization: Bearer caa1b60d-61ff-4cd8-a4e1-2d18c8696de0
```

3.11.2 Response: accounts/{`AccountId`}/transactions resource

```
HTTP/1.1 200 OK
Content-Type: application/json
x-fapi-interaction-id: 942a7ee7-d29a-45aa-93b7-c5f292d86602
{
  "Data": {
    "AccountId": "f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115",
    "Transaction": [
      {
        "TransactionId": "668f2fc8-7aa8-411b-bb9f-7571a90e7512",
        "TransactionReference": "1852efce-bedc-4fda-ba51-0f76c9137f91",
        "CreditDebitIndicator": "Debit",
        "Status": "Booked",
        "TransactionMutability": "",
        "BookingDateTime": "2016-01-04T00:00:00.000Z",
        "ValueDateTime": "2016-01-04T00:00:00.000Z",
        "TransactionInformation": "Foo Group ",
        "Amount": {
          "Amount": "41.10",
          "Currency": "SAR"
        },
        "BankTransactionCode": {
          "Code": "CustomerCardTransactions",
          "SubCode": "CashWithdrawal"
        },
        "ProprietaryBankTransactionCode": {
          "Code": ""
        }
      },
      {
        "TransactionId": "05b6bfde-ce5a-48e1-a448-66d75518f1e8",
        "TransactionReference": "b5a6a869-730f-449d-badf-14ebf3980147",
        "CreditDebitIndicator": "Debit",
        "Status": "Booked",
        "TransactionMutability": "",
        "BookingDateTime": "2016-01-19T00:00:00.000Z",
        "ValueDateTime": "2016-01-19T00:00:00.000Z",
        "TransactionInformation": "Bar Holding",
        "Amount": {
          "Amount": "32.40",
          "Currency": "SAR"
        },
        "BankTransactionCode": {
          "Code": "CustomerCardTransactions",
          "SubCode": "CashWithdrawal"
        },
        "ProprietaryBankTransactionCode": {
          "Code": ""
        }
      }
    ]
  },
  "Links": {
    "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/accoun
  },
  "Meta": {
    "TotalPages": 1,
    "FirstAvailableDateTime": "2022-01-01T10:40:00+02:00",
    "LastAvailableDateTime": "2022-12-31T10:40:00+02:00"
  }
}
```

4. Further Examples

4.1 The TPP Queries the Account Access Consent Resource for the Status after a PSU has Authorized the Consent with Their PASP

4.1.1 Request: account-access-consents/{`ConsentId`} resource

```
GET /open-banking/account-information/2022.11.01-final-errata2/account-access-consents/aac-1a672e83-d1e5-42bc-b8e
Host: rs1.lab.api.openbanking.sa
Content-Type: application/json
x-fapi-interaction-id: 2e974f01-d111-4078-9a19-7a9b385e637c
Authorization: Bearer e6156449-6f27-4c42-aa5b-36602f73eac9
```

4.1.2 Response: account-access-consents/{`ConsentId`} resource

```
HTTP/1.1 200 OK
Content-Type:application/json
x-fapi-interaction-id: 2e974f01-d111-4078-9a19-7a9b385e637c
{
    "Data": {
        "Permissions": [
            "ReadAccountsBasic",
            "ReadAccountsDetail",
            "ReadBalances",
            "ReadBeneficiariesBasic",
            "ReadBeneficiariesDetail",
            "ReadDirectDebits",
            "ReadTransactionsBasic",
            "ReadTransactionsCredits",
            "ReadTransactionsDebits",
            "ReadTransactionsDetail",
            "ReadProducts",
            "ReadStandingOrdersDetail",
            "ReadProducts",
            "ReadStandingOrdersDetail",
            "ReadStatementsDetail",
            "ReadParty",
            "ReadPartyPSU"
        ],
        "TransactionFromDateTime": "2016-01-01T10:40:00+02:00",
        "TransactionToDateTime": "2025-12-31T10:40:00+02:00",
        "ConsentId": "aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd",
        "Status": "Authorized",
        "CreationDateTime": "2022-08-07T08:01:41.344Z",
        "StatusUpdateDateTime": "2022-08-07T09:22:13.173Z"
    },
    "Risk": {},
    "Links": {
        "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/ac
    },
    "Meta": {}
}
```

4.2 The TPP Requests a List of Accounts Using an Expired Access Token

4.2.1 Request: accounts resource

```
GET /open-banking/account-information/2022.11.01-final-errata2/accounts HTTP/1.1
Host: rs1.lab.api.openbanking.sa
Content-Type: application/json
x-fapi-interaction-id: 9a371b79-4e79-4d7d-a77d-380c528ab8c0
Authorization: Bearer caa1b60d-61ff-4cd8-a4e1-2d18c8696de0
```

4.2.2 Response: 403 Forbidden

```
HTTP/1.1 401 Unauthorized
Content-Type: application/json
x-fapi-interaction-id: 9a371b79-4e79-4d7d-a77d-380c528ab8c0
{
  "Errors": [
    {
      "Code": "KSAOB.AccessToken.Unauthorized",
      "Message": "max_age_exceeded: Token has expired",
      "Path": "Authorization",
      "Url": "https://developer.openbanking.sa/api-errors/401"
    }
  ]
}
```

4.3 Webhooks

4.3.1 The TPP Creates an Account Consent Request with a Webhook Subscription

4.3.1.1 Request: Data Permissions and Webhook Subscription

```
POST /open-banking/account-information/2022.11.01-final-errata2/account-access-consents HTTP/1.1
Host: rs1.lab.api.openbanking.sa
Accept: application/json
Content-Type: application/json
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Authorization: Bearer ad297304-1057-4c68-9e76-a96f300a27f1
{
  "Data": {
    "Permissions": [
      "ReadAccountsBasic",
      "ReadAccountsDetail"
    ],
    "ExpirationDateTime": "2022-01-10T10:40:00+02:00",
    "TransactionFromDateTime": "2022-01-01T10:40:00+02:00",
    "TransactionToDateTime": "2022-12-31T10:40:00+02:00"
  },
  "Subscription": {
    "Webhook": {
      "Url": "https://api.tpp.com/webhook/callbackUrl",
      "IsActive": false
    }
  }
}

```

4.3.1.2 Response: PASP Accepts the TPP Account Consent Request with Webhook Subscription and Returns a ConsentId and Status

```
HTTP/1.1 201 Created
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Content-Type: application/json
{
  "Data": {
    "Permissions": [
      "ReadAccountsBasic",
      "ReadAccountsDetail"
    ],
    "TransactionFromDateTime": "2022-01-01T10:40:00+02:00",
    "TransactionToDateTime": "2022-12-31T10:40:00+02:00",
    "ConsentId": "aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd",
    "Status": "AwaitingAuthorization",
    "CreationDateTime": "2022-08-07T08:01:41.344Z",
    "StatusUpdateDateTime": "2022-08-07T08:01:41.344Z"
  },
  "Subscription": {
    "Webhook": {
      "Url": "https://api.tpp.com/webhook/callbackUrl",
      "IsActive": false
    }
  },
  "Links": {
    "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/accour
  },
  "Meta": {}
}
```

4.3.2 The TPP updates a Webhook Subscription preference with the PASP

4.3.2.1 Request: Activate Webhook events

```
PATCH /open-banking/account-information/2022.11.01-final-errata2/account-access-consents/aac-1a672e83-d1e5-42bc-
Host: rs1.lab.api.openbanking.sa
Content-Type: application/json
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Authorization: Bearer ad297304-1057-4c68-9e76-a96f300a27f1
```

```
{
  "Subscription": {
    "Webhook": {
      "IsActive": true
    }
  }
}
```

4.3.2.2 Response: Webhook events activated

```
HTTP/1.1 204 No Content
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
```

4.3.3 The TPP unsubscribes their Webhook Subscription with the PASP

4.3.3.1 Request: De-Activate Webhook events

```
PATCH /open-banking/account-information/2022.11.01-final-errata2/account-access-consents/aac-1a672e83-d1e5-42bc-
Host: rs1.lab.api.openbanking.sa
Content-Type: application/json
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
Authorization: Bearer ad297304-1057-4c68-9e76-a96f300a27f1
{
  "Subscription": {
    "Webhook": {
      "IsActive": false
    }
  }
}
```

4.3.3.2 Response: Webhook events de-activated

```
HTTP/1.1 204 No Content
x-fapi-interaction-id: 3424a379-8274-4686-99bd-f420d08acead
```

4.3.4 The TPP receives data from the PASP (specific to the consent and permissions) via its webhook

4.3.4.1 The PASP generates a Self Signed JWT Authorization Token for Client Authentication with the TPP.

This JWT Authorization Token MUST be set in the Authorization Header.

```
{
  "alg": "PS256",
  "typ": "JOSE",
  "cty": "json",
  "kid":  "e1be6bf3-76e6-4e53-92b9-c46423757ab1"
}
.
{
 "iss": "https://openbanking.masrif-ahmar.sa",
 "sub": "e75c26bf-1682-401a-a227-ec125f6636ab",
 "aud": "https://api.tpp.com/webhook/callbackUrl",
 "exp": 1661378066,
 "iat": 1661378036,
 "nbf": 1661378036,
 "jti": "274aa39d-d77a-46a9-b832-b2ced47919dd"
}
```

```
.
<<signature>>
```

4.3.4.2 Request: PASP publishes signed/encrypted data to the registered Webhook Url provided by the TPP

The example below shows a signed and encrypted payload with the JWT Authorization Token set in the Authorization Header.

```
POST /webhook/callbackUrl HTTP/1.1
Host: api.tpp.com
x-fapi-interaction-id: 77b0e830-b095-4c6c-94e8-20f83eaa799f
Content-Type: application/jwt
Date: Wed, 24 Aug 2022 07:28:00 AST
Authorization: Bearer eyJhbGciO9.eyJzdWImlhdCI6MTUxNjIzOTAyMn0.iOeN9eg

<<jwe>>

```

Here, `<<jwe>>` is a signed and encrypted payload. The inner JWS has the structure below:

```
{
  "alg": "PS256",
  "kid": "e1be6bf3-76e6-4e53-92b9-c46423757ab1"
}
.
{
  "iss": "string",
  "exp": 1664950125,
  "nbf": 1664950125,
  "aud": [
    "6uC8HSQ8C59SDSw43Cdm9YWxxjJmDV"
  ],
  "iat": 1661378036,
  "message": {
    "Data": {
      "AccountId": "f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115",
      "Account": [
        {
          "Currency": "SAR",
          "Status": "Active",
          "AccountIdentifiers": [
            {
              "Name": "Account 1",
              "IdentificationType": "KSAOB.IBAN",
              "Identification": "00003130000001"
            }
          ]
        }
      ]
    },
    "Links": {
      "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/acc
    },
    "EventMeta": {
      "EventDateTime": "2022-08-24T07:28:00.556Z",
      "EventResource": "accounts",
      "EventType": "KSAOB.Resource.Created",
      "ConsentId": "aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd"
    }
  }
}
.
<<signature>>
```

4.3.4.3 Response: TPP validates the Self Signed JWT Authorization Token from PASP, stores data, and acknowledges a successful response to the PASP

```
HTTP/1.1 202 Accepted
x-fapi-interaction-id: 77b0e830-b095-4c6c-94e8-20f83eaa799f
```

4.3.5 Webhook Payload for a single resource collection associated with a single event type

The following non-normative example illustrates a PASP webhook payload for a collection of account ID transactions of the same event type: `KSAOB.Resource.Created`

```
{
  "alg": "PS256",
  "kid": "e1be6bf3-76e6-4e53-92b9-c46423757ab1"
}
.
{
  "iss": "string",
  "exp": 1664950125,
  "nbf": 1664950125,
  "aud": [
    "6uC8HSQ8C59SDSw43Cdm9YWxxjJmDV"
  ],
  "iat": 1661378036,
  "message": {
    "Data": {
      "AccountId": "f91d07d0-6d8f-4e0e-9fb4-0ac61f84d115",
      "Transaction": [
        {
          "TransactionId": "668f2fc8-7aa8-411b-bb9f-7571a90e7512",
          "TransactionReference": "1852efce-bedc-4fda-ba51-0f76c9137f91",
          "CreditDebitIndicator": "Debit",
          "Status": "Booked",
          "TransactionMutability": "",
          "BookingDateTime": "2022-08-24T07:27:00.556Z",
          "ValueDateTime": "2022-08-24T07:27:00.556Z",
          "TransactionInformation": "Foo Group ",
          "Amount": {
            "Amount": "41.10",
            "Currency": "SAR"
          },
          "BankTransactionCode": {
            "Code": "CustomerCardTransactions",
            "SubCode": "CashWithdrawal"
          },
          "ProprietaryBankTransactionCode": {
            "Code": ""
          }
        },
        {
          "TransactionId": "05b6bfde-ce5a-48e1-a448-66d75518f1e8",
          "TransactionReference": "b5a6a869-730f-449d-badf-14ebf3980147",
          "CreditDebitIndicator": "Debit",
          "Status": "Booked",
          "TransactionMutability": "",
          "BookingDateTime": "2022-08-24T07:28:00.556Z",
          "ValueDateTime": "2022-08-24T07:28:00.556Z",
          "TransactionInformation": "Bar Holding",
          "Amount": {
            "Amount": "32.40",
            "Currency": "SAR"
          },
          "BankTransactionCode": {
            "Code": "CustomerCardTransactions",
            "SubCode": "CashWithdrawal"
          },
          "ProprietaryBankTransactionCode": {
            "Code": ""
          }
        }
      ]
    },
    "Links": {
      "Self": "https://rs1.lab.api.openbanking.sa/open-banking/account-information/2022.11.01-final-errata2/acco
    },
    "EventMeta": {
      "EventDateTime": "2022-08-24T08:28:00.556Z",
      "EventResource": "transactions",
      "EventType": "KSAOB.Resource.Created",
      "ConsentId": "aac-1a672e83-d1e5-42bc-b8e1-60a490ec52fd"
    }
  }
}
.
<<signature>>
```

5. OpenAPI Specification

See Account Information API Specification

6. Notes

- IBAN
  - Where Data/Account/AccountIdentifiers.IdentificationType is KSAOB.IBAN, the Identification field SHOULD contain the full IBAN
- Passport, Driving Permit, IDCard, Residence Permit
  - Where Data/Party/VerifiedClaims/Verification/Evidence[DocumentDetails.Type] is set to any one of these enumerated values, then the DocumentNumber field SHOULD contain the actual number of the document type specified.
- ProprietaryBankTransactionCodes
  - This code is mandatory when the BenefeciaryCode with code specifying the Domain, Family, and SubFamily as per External Codes ISO20022 is absent. This code is a proprietary code from the PASP and does not have a defined code list.