

Account Information Service Provider (AISP)
Payment Instrument Issuing Service Provider (PIISP)

APIs documentation summary



Table of Contents

1.	AISP - Read Account List	3
2.	AISP - Read Account Details	3
3.	AISP - Read Balance	4
4.	AISP - Read Transaction list of an account	4
5.	AISP - Read Transaction Details	5
6.	AISP - Create Consent	5
7.	AISP - Delete Consent	6
8.	AISP - Get Consent Request	6
9.	AISP - Get Consent Authorisation Sub-Resources Request	6
10.	AISP - Start the authorisation process for a consent	7
11.	AISP - Read the SCA status of the consent authorisation	7
12.	AISP - Update PSU Data for consents	8
13.	AISP - Consent Status Request	8
14.	PIISP – Confirmation of Funds Request	9
15.	PIISP - Create Funds Confirmation Consent	9
16.	PIISP - Delete Funds Confirmation Consent	9
17.	PIISP - Get Funds Confirmation Consent Request	10
18.	PIISP - Get Funds Confirmation Consent Authorisation Sub-Resources Request	10
19.	PIISP - Start the authorisation process for a Funds Confirmation consent	11
20.	PIISP - Read the SCA status of the consent authorisation	11
21.	PIISP - Update PSU Data for Funds Confirmation consents	11
22.	PIISP - Funds Confirmation Consent status request	12

23. Accounts Annexes	12
24. Consent Annexes	14

1. AISP - Read Account List

1.1. Resource

GET /v1/accounts

1.2. Description

Read the identifiers of the available payment account together with booking balance information, depending on the consent granted. It is assumed that a consent of the PSU to this access is already given and stored on the ASPSP system. The addressed list of accounts depends then on the PSU ID and the stored consent addressed by consentId, respectively the OAuth2 access token. Returns all identifiers of the accounts, to which an account access has been granted to through the /consents endpoint by the PSU. In addition, relevant information about the accounts and hyperlinks to corresponding account information resources are provided if a related consent has been already granted. Remark: Note that the /consents endpoint optionally offers to grant an access on all available payment accounts of a PSU. In this case, this endpoint will deliver the information about all available payment accounts of the PSU at this ASPSP.

1.3. Parameters

1.3.1. Header

c.f. annex 23.1

1.3.2. Query

c.f. annex 23.2

2. AISP - Read Account Details

2.1. Resource

GET /v1/accounts/{account-id}

2.2. Description

Reads details about an account, with balances where required. It is assumed that a consent of the PSU to this access is already given and stored on the ASPSP system. The addressed details of this account depends then on the stored consent addressed by consentId, respectively the OAuth2 access token. NOTE: The account-id can represent a multicurrency account. In this case the currency code is set to "XXX". Give detailed information about the addressed account. Give detailed information about the addressed account together with balance information.

2.3. Parameters

2.3.1. Header

c.f. annex 23.1

2.3.2. Query

c.f. annex 23.2

2.3.3. Path

c.f. annex 23.4

3. AISP - Read Balance

3.1. Resource

GET /v1/accounts/{account-id}/balances

3.2. Description

Reads account data from a given account addressed by "account-id". Remark: This account-id can be a tokenised identification due to data protection reason since the path information might be logged on intermediary servers within the ASPSP sphere. This account-id then can be retrieved by the "GET Account List" call. The account-id is constant at least throughout the lifecycle of a given consent.

3.3. Parameters

3.3.1. Header

c.f. annex 23.1

3.3.2. Path

c.f. annex 23.4

4. AISP - Read Transaction list of an account

4.1. Resource

GET /v1/accounts/{account-id}/transactions

4.2. Description

Read transaction reports or transaction lists of a given account addressed by "account-id", depending on the steering parameter "bookingStatus" together with balances. For a given account, additional parameters are e.g. the attributes "dateFrom" and "dateTo". The ASPSP might add balance information, if transaction lists without balances are not supported.

4.3. Parameters

4.3.1. Header

c.f. annex 23.1

4.3.2. Query

c.f. annex 23.3

4.3.3. Path

c.f. annex 23.5

5. AISP - Read Transaction Details

5.1. Resource

GET /v1/accounts/{account-id}/transactions/{resourceId}

5.2. Description

Reads transaction details from a given transaction addressed by "resourceId" on a given account addressed by "account-id". This call is only available on transactions as reported in a JSON format.

5.3. Parameters

5.3.1. Header

c.f. annex 23.1

5.3.2. Path

c.f. annex 23.6

6. AISP - Create Consent

6.1. Resource

POST /v1/consents

6.2. Description

This method creates a consent resource, defining access rights to dedicated accounts of a given PSU-ID. These accounts are addressed explicitly in the method as parameters as a core function. Side Effects When this Consent Request is a request where the "recurringIndicator" equals "true", and if it exists already a former consent for recurring access on account information for the addressed PSU, then the former consent automatically expires as soon as the new consent request is authorised by the PSU. Optional Extension: As an option, an ASPSP might optionally accept a specific access right on the access on all psd2 related services for all available accounts. As another option an ASPSP might optionally also accept a command, where only access rights are inserted without mentioning the addressed account. The relation to accounts is then handled afterwards between PSU and ASPSP. This option is not supported for the Embedded SCA Approach. As a last option, an ASPSP might in addition accept a command with access rights.

6.3. Parameters

6.3.1. Header

c.f. annex 24.1

6.3.2. *Body*

c.f. annex 24.5

7. AISP - Delete Consent

7.1. Resource

DELETE /v1/consents/{consentId}

7.2. Description

The TPP can delete an account information consent object if needed.

7.3. Parameters

7.3.1. *Header*

c.f. annex 24.2

7.3.2. *Path*

c.f. annex 24.3

8. AISP - Get Consent Request

8.1. Resource

GET /v1/consents/{consentId}

8.2. Description

Returns the content of an account information consent object. This is returning the data for the TPP especially in cases, where the consent was directly managed between ASPSP and PSU e.g. in a re-direct SCA Approach.

8.3. Parameters

8.3.1. *Header*

c.f. annex 24.2

8.3.2. *Path*

c.f. annex 24.3

9. AISP - Get Consent Authorisation Sub-Resources Request

9.1. Resource

GET /v1/consents/{consentId}/authorisations

9.2. Description

Returns a list of all authorisation subresources IDs which have been created. This function returns an array of hyperlinks to all generated authorisation sub-resources.

9.3. Parameters

9.3.1. Header

c.f. annex 24.2

9.3.2. Path

c.f. annex 24.3

10. AISP - Start the authorisation process for a consent

10.1. Resource

POST /v1/consents/{consentId}/authorisations

10.2. Description

Create an authorisation sub-resource and start the authorisation process. The message might in addition transmit authentication and authorisation related data. This method is iterated n times for a n times SCA authorisation in a corporate context, each creating an own authorisation sub-endpoint for the corresponding PSU authorizing the transaction. The ASPSP might make the usage of this access method unnecessary in case of only one SCA process needed, since the related authorisation resource might be automatically created by the ASPSP after the submission of the payment data with the first POST payments/{payment-product} call. The start authorisation process is a process which is needed for creating a new authorisation or cancellation sub-resource.

10.3. Parameters

10.3.1. Header

c.f. annex 24.2

10.3.2. Path

c.f. annex 24.3

11. AISP - Read the SCA status of the consent authorisation

11.1. Resource

GET /v1/consents/{consentId}/authorisations{authorisationId}

11.2. Description

This method returns the SCA status of a consent initiation's authorisation sub-resource.

11.3. Parameters

11.3.1. Header

c.f. annex 24.2

11.3.2. Path

c.f. annex 24.4

12. AISP - Update PSU Data for consents

12.1. Resource

PUT /v1/consents/{consentId}/authorisations

12.2. Description

This method updates PSU data on the consents resource if needed. It may authorize a consent within the Embedded SCA Approach where needed. Independently from the SCA Approach it supports e.g. the selection of the authentication method and a non-SCA PSU authentication. This method updates PSU data on the cancellation authorisation resource if needed.

12.3. Parameters

12.3.1. Header

c.f. annex 24.1

12.3.2. Path

c.f. annex 24.4

13. AISP - Consent Status Request

13.1. Resource

GET /v1/consents/{consentId}/status

13.2. Description

This method allows to check the status of an account information consent resource.

13.3. Parameters

13.3.1. Header

c.f. annex 24.2

13.3.2. Path

c.f. annex 24.3

14. PIISP – Confirmation of Funds Request

14.1. Resource

POST /v1/funds-confirmations

14.2. Description

This method creates a confirmation of funds request at the ASPSP. Checks whether a specific amount is available at point of time of the request on an account linked to a given tuple card issuer(TPP)/card number, or addressed by IBAN and TPP respectively

14.3. Parameters

14.3.1. Header

c.f. annex 24.2

14.3.2. Body

c.f. annex 23.7

15. PIISP - Create Funds Confirmation Consent

15.1. Resource

POST /v1/consents/confirmation-of-funds

15.2. Description

This method creates a confirmation of funds consent resource at the ASPSP regarding confirmation of funds access to an account specified in this request.

15.3. Parameters

15.3.1. Header

c.f. annex 24.1

15.3.2. Body

c.f. annex 24.5

16. PIISP - Delete Funds Confirmation Consent

16.1. Resource

DELETE /v1/consents/confirmation-of-funds/{consentId}

16.2. Description

The TPP can delete a confirmation of funds consent object if needed.

16.3. Parameters

16.3.1. Header

c.f. annex 24.2

16.3.2. Path

c.f. annex 24.3

17. PIISP - Get Funds Confirmation Consent Request

17.1. Resource

GET /v1/consents/confirmation-of-funds/{consentId}

17.2. Description

Returns the content of an account information consent object. This is returning the data for the TPP especially in cases, where the consent was directly managed between ASPSP and PSU e.g. in a re-direct SCA Approach

17.3. Parameters

17.3.1. Header

c.f. annex 24.2

17.3.2. Path

c.f. annex 24.3

18. PIISP - Get Funds Confirmation Consent Authorisation Sub-Resources Request

18.1. Resource

GET /v1/consents/confirmation-of-funds/{consentId}/authorisations

18.2. Description

Returns a list of all authorisation subresources IDs which have been created. This function returns an array of hyperlinks to all generated authorisation sub-resources.

18.3. Parameters

18.3.1. Header

c.f. annex 24.2

18.3.2. Path

c.f. annex 24.3

19. PIISP - Start the authorisation process for a Funds Confirmation consent

19.1. Resource

POST /v1/consents/confirmation-of-funds/{consentId}/authorisations

19.2. Description

Create an authorisation sub-resource and start the authorisation process. The message might in addition transmit authentication and authorisation related data. This method is iterated n times for a n times SCA authorisation in a corporate context, each creating an own authorisation sub-endpoint for the corresponding PSU authorizing the transaction. The ASPSP might make the usage of this access method unnecessary in case of only one SCA process needed, since the related authorisation resource might be automatically created by the ASPSP after the submission of the payment data with the first POST payments/{payment-product} call. The start authorisation process is a process which is needed for creating a new authorisation or cancellation sub-resource.

19.3. Parameters

19.3.1. Header

c.f. annex 24.2

19.3.2. Path

c.f. annex 24.3

20. PIISP - Read the SCA status of the consent authorisation

20.1. Resource

GET /v1/consents/confirmation-of-funds/{consentId}/authorisations/{authorisationId}

20.2. Description

This method returns the SCA status of a consent initiation's authorisation sub-resource.

20.3. Parameters

20.3.1. Header

c.f. annex 24.2

20.3.2. Path

c.f. annex 24.4

21. PIISP - Update PSU Data for Funds Confirmation consents

21.1. Resource

PUT /v1/consents/confirmation-of-funds/{consentId}/authorisations/{authorisationId}

21.2. Description

This method updates PSU data on the consents resource if needed. It may authorize a consent within the Embedded SCA Approach where needed. Independently from the SCA Approach it supports e.g. the selection of the authentication method and a non-SCA PSU authentication. This method updates PSU data on the cancellation authorisation resource if needed.

21.3. Parameters

21.3.1. Header

c.f. annex 24.1

21.3.2. Path

c.f. annex 24.4

22. PIISP - Funds Confirmation Consent status request

22.1. Resource

GET /v1/consents/confirmation-of-funds/{consentId}/status

22.2. Description

This method allows to check the status of an account information consent resource.

22.3. Parameters

22.3.1. Header

c.f. annex 24.2

22.3.2. Path

c.f. annex 24.3

23. Accounts Annexes

23.1. Accounts Header Parameters

Name	Condition	Description	Format
X-Request-ID	required	ID of the request, unique to the call, as determined by the initiating party.	string(uuid)
Digest	optional	Is contained if and only if the "Signature" element is contained in the header of the request.	string
Signature	optional	A signature of the request by the TPP on application level. This might be mandated by ASPSP.	string
TPP-Signature-Certificate	optional	The certificate used for signing the request, in base64 encoding. Must be contained if a signature is contained.	string(byte)
Consent-ID	required	This then contains the consentId of the related AIS consent, which was performed prior to this payment initiation.	string
PSU-IP-Address	optional	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP. It shall be contained if and only if this request was actively initiated by the PSU.	string(ipv4)

PSU-IP-Port	optional	The forwarded IP Port header field consists of the corresponding HTTP request IP Port field between PSU and TPP, if available.	string
PSU-Accept	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Charset	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Encoding	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Language	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-User-Agent	optional	The forwarded Agent header field of the HTTP request between PSU and TPP, if available.	string
PSU-Http-Method	optional	HTTP method used at the PSU ? TPP interface, if available. Valid values are: * GET * POST * PUT * PATCH * DELETE	string
PSU-Device-ID	optional	UUID (Universally Unique Identifier) for a device, which is used by the PSU, if available. UUID identifies either a device or a device dependant application installation. In case of an installation identification this ID need to be unaltered until removal from device.	string(uuid)
PSU-Geo-Location	optional	The forwarded Geo Location of the corresponding http request between PSU and TPP if available.	string
Accept	optional	application/json	

23.2. Accounts Query Parameters

Name	Condition	Description	Format
withBalance	optional	If contained, this function reads the list of accessible payment accounts including the booking balance, if granted by the PSU in the related consent and available by the ASPSP. This parameter might be ignored by the ASPSP.	boolean

23.3. Transactions Query Parameters

Name	Condition	Description	Format
dateFrom	optional	Conditional: Starting date (inclusive the date dateFrom) of the transaction list, mandated if no delta access is required. For booked transactions, the relevant date is the booking date. For pending transactions, the relevant date is the entry date, which may not be transparent neither in this API nor other channels of the ASPSP.	string(date)
dateTo	optional	End date (inclusive the data dateTo) of the transaction list, default is "now" if not given. Might be ignored if a delta function is used. For booked transactions, the relevant date is the booking date. For pending transactions, the relevant date is the entry date, which may not be transparent neither in this API nor other channels of the ASPSP.	string(date)
entryReferenceFrom	optional	This data attribute is indicating that the AISP is in favour to get all transactions after the transaction with identification entryReferenceFrom alternatively to the above defined period. This is a implementation of a delta access. If this data element is contained, the entries "dateFrom" and "dateTo" might be ignored by the ASPSP if a delta report is supported. Optional if supported by API provider.	string
bookingStatus	required	Permitted codes are * "booked", * "pending" and * "both" "booked" shall be supported by the ASPSP. To support the "pending" and "both" feature is optional for the ASPSP, Error code if not supported in the online banking frontend	string



deltaList	optional	This data attribute is indicating that the AISP is in favour to get all transactions after the last report access for this PSU on the addressed account. This is another implementation of a delta access-report. This delta indicator might be rejected by the ASPSP if this function is not supported. Optional if supported by API provider	boolean
withBalance	optional	If contained, this function reads the list of accessible payment accounts including the booking balance, if granted by the PSU in the related consent and available by the ASPSP. This parameter might be ignored by the ASPSP.	boolean

23.4. Accounts Path Parameters

Name	Condition	Description	Format
Account-id	required	This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call. The account-id is the "id" attribute of the account structure. Its value is constant at least throughout the lifecycle of a given consent.	string

23.5. Transactions Path Parameters

Name	Condition	Description	Format
Account-id	required	This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call. The account-id is the "id" attribute of the account structure. Its value is constant at least throughout the lifecycle of a given consent.	string

23.6. Transactions Path Parameters with Resource

Name	Condition	Description	Format
Account-id	required	This identification is denoting the addressed account. The account-id is retrieved by using a "Read Account List" call. The account-id is the "id" attribute of the account structure. Its value is constant at least throughout the lifecycle of a given consent.	string
resourceId	required	This identification is given by the attribute resourceId of the corresponding entry of a transaction list.	string

23.7. Confirmation of Funds request Body

Name	Condition	Description	Format
cardNumber	optional	Card Number of the card issued by the PIISP. Should be delivered if available.	String
account	mandatory	PSU's account number.	Account Reference
payee	optional	The merchant where the card is accepted as an information to the PSU.	Max70Text
instructedAmount	mandatory	Transaction amount to be checked within the funds check mechanism.	Amount

24. Consent Annexes

24.1. Consent Header Parameters

Name	Condition	Description	Format
X-Request-ID	required	ID of the request, unique to the call, as determined by the initiating party.	string(uuid)
Digest	optional	Is contained if and only if the "Signature" element is contained in the header of the request.	string
Signature	optional	A signature of the request by the TPP on application level. This might be mandated by ASPSP.	string
TPP-Signature-Certificate	optional	The certificate used for signing the request, in base64 encoding. Must be contained if a signature is contained.	string(byte)

PSU-ID	optional	Client ID of the PSU in the ASPSP client interface. Might be mandated in the ASPSP's documentation. Is not contained if an OAuth2 based authentication was performed in a pre-step or an OAuth2 based SCA was performed in an preceding AIS service in the same session.	string
PSU-ID-Type	optional	Type of the PSU-ID, needed in scenarios where PSUs have several PSU-IDs as access possibility.	string
PSU-Corporate-ID	optional	Might be mandated in the ASPSP's documentation. Only used in a corporate context.	string
PSU-Corporate-ID-Type	optional	Might be mandated in the ASPSP's documentation. Only used in a corporate context.	string
TPP-Redirect-Preferred	optional	If it equals "true", the TPP prefers a redirect over an embedded SCA approach. If it equals "false", the TPP prefers not to be redirected for SCA. The ASPSP will then choose between the Embedded or the Decoupled SCA approach, depending on the choice of the SCA procedure by the TPP/PSU. If the parameter is not used, the ASPSP will choose the SCA approach to be applied depending on the SCA method chosen by the TPP/PSU.	string
TPP-Redirect-URI	optional	URI of the TPP, where the transaction flow shall be redirected to after a Redirect. Mandated for the Redirect SCA Approach (including OAuth2 SCA approach), specifically when TPP-Redirect-Preferred equals "true". It is recommended to always use this header field. **Remark for Future:** This field might be changed to mandatory in the next version of the specification.	string(uri)
TPP-Nok-Redirect-URI	optional	If this URI is contained, the TPP is asking to redirect the transaction flow to this address instead of the TPP-Redirect-URI in case of a negative result of the redirect SCA method. This might be ignored by the ASPSP.	string(uri)
TPP-Explicit-Authorisation-Preferred	optional	If it equals "true", the TPP prefers to start the authorisation process separately, e.g. because of the usage of a signing basket. This preference might be ignored by the ASPSP, if a signing basket is not supported as functionality. If it equals "false" or if the parameter is not used, there is no preference of the TPP. This especially indicates that the TPP assumes a direct authorisation of the transaction in the next step, without using a signing basket.	string
PSU-IP-Address	optional	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP. It shall be contained if and only if this request was actively initiated by the PSU.	string(ipv4)
PSU-IP-Port	optional	The forwarded IP Port header field consists of the corresponding HTTP request IP Port field between PSU and TPP, if available.	string
PSU-Accept	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Charset	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Encoding	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Language	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-User-Agent	optional	The forwarded Agent header field of the HTTP request between PSU and TPP, if available.	string
PSU-Http-Method	optional	HTTP method used at the PSU ? TPP interface, if available. Valid values are: * GET * POST * PUT * PATCH * DELETE	string
PSU-Device-ID	optional	UUID (Universally Unique Identifier) for a device, which is used by the PSU, if available. UUID identifies either a device or a device dependant application installation. In case of an installation identification this ID need to be unaltered until removal from device.	string(uuid)



PSU-Geo-Location	optional	The forwarded Geo Location of the corresponding http request between PSU and TPP if available.	string
Accept	optional	application/json	
Content-Type	optional	application/json	

24.2. Consent Header Parameters

Name	Condition	Description	Format
X-Request-ID	required	ID of the request, unique to the call, as determined by the initiating party.	string(uuid)
Digest	optional	Is contained if and only if the "Signature" element is contained in the header of the request.	string
Signature	optional	A signature of the request by the TPP on application level. This might be mandated by ASPSP.	string
TPP-Signature-Certificate	optional	The certificate used for signing the request, in base64 encoding. Must be contained if a signature is contained.	string(byte)
PSU-IP-Address	optional	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP. It shall be contained if and only if this request was actively initiated by the PSU.	string(ipv4)
PSU-IP-Port	optional	The forwarded IP Port header field consists of the corresponding HTTP request IP Port field between PSU and TPP, if available.	string
PSU-Accept	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Charset	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Encoding	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-Accept-Language	optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.	string
PSU-User-Agent	optional	The forwarded Agent header field of the HTTP request between PSU and TPP, if available.	string
PSU-Http-Method	optional	HTTP method used at the PSU ? TPP interface, if available. Valid values are: * GET * POST * PUT * PATCH * DELETE	string
PSU-Device-ID	optional	UUID (Universally Unique Identifier) for a device, which is used by the PSU, if available. UUID identifies either a device or a device dependant application installation. In case of an installation identification this ID need to be unaltered until removal from device.	string(uuid)
PSU-Geo-Location	optional	The forwarded Geo Location of the corresponding http request between PSU and TPP if available.	string
Accept	optional	application/json or application/xml	

24.3. Consent Path Parameters

Name	Condition	Description	Format
consentId	required	ID of the corresponding consent object as returned by an Account Information Consent Request.	string

24.4. Consent Path Parameters

Name	Condition	Description	Format
consentId	required	ID of the corresponding consent object as returned by an Account Information Consent Request.	string
authorisationId	required	Resource identification of the related SCA.	string

24.5. Consent Body Parameters

Name	Condition	Description	Format
body	optional	Request body for a consent request	json



