THE Berlin GROUP

A EUROPEAN STANDARDS INITIATIVE

NextGenPSD2 XS2A Framework

Implementation Guidelines

Extended Services

Confirmation of Funds Consent

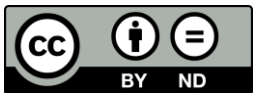
Version 2.0

01. March 2019

License Notice

This Specification has been prepared by the Participants of the Joint Initiative pan-European PSD2-Interface Interoperability* (hereafter: Joint Initiative). This Specification is published by the Berlin Group under the following license conditions:

- "Creative Commons Attribution-NoDerivatives 4.0 International Public License"

  This means that the Specification can be copied and redistributed in any medium or format for any purpose, even commercially, and when shared, that appropriate credit must be given, a link to the license must be provided, and indicated if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use. In addition, if you remix, transform, or build upon the Specification, you may not distribute the modified Specification.

- Implementation of certain elements of this Specification may require licenses under third party intellectual property rights, including without limitation, patent rights. The Berlin Group or any contributor to the Specification is not, and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights.
- The Specification, including technical data, may be subject to export or import regulations in different countries. Any user of the Specification agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import (parts of) the Specification.

---

* The 'Joint Initiative pan-European PSD2-Interface Interoperability' brings together participants of the Berlin Group with additional European banks (ASPSPs), banking associations, payment associations, payment schemes and interbank processors.

Contents

1 Introduction

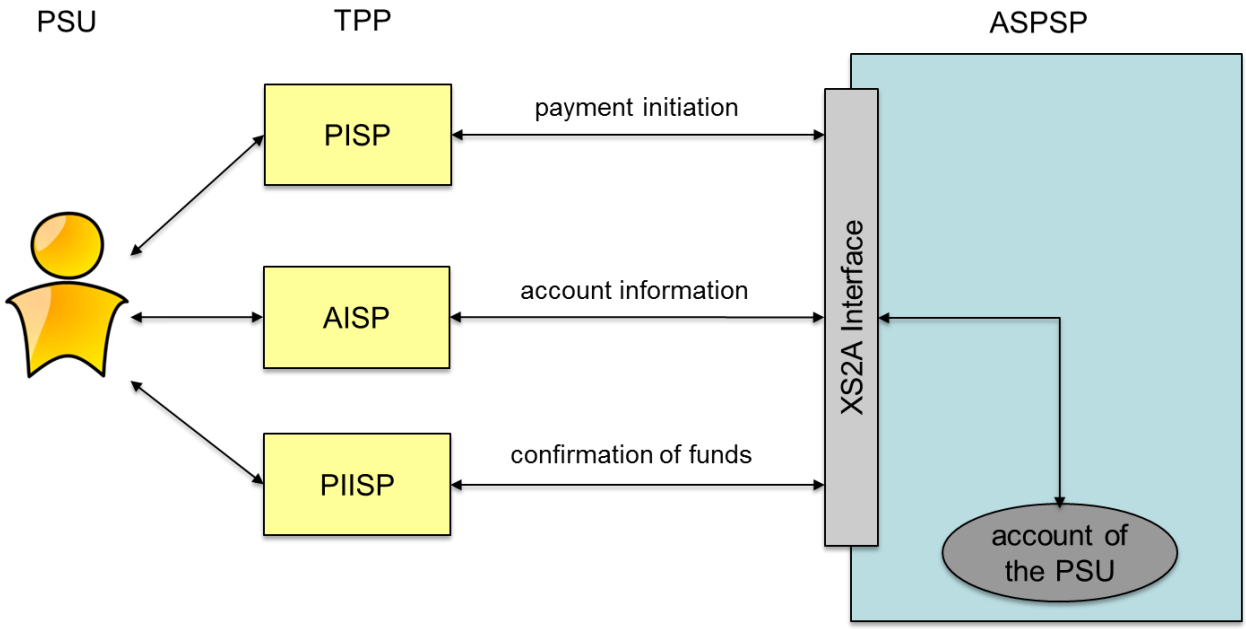
1.1 Background

With [PSD2] the European Union has published a new directive on payment services in the internal market. Member States had to adopt this directive into their national law until 13th of January 2018.

Among others [PSD2] contains regulations of new services to be operated by so called Third Party Payment Service Providers (TPP) on behalf of a Payment Service User (PSU). These new services are

- Payment Initiation Service (PIS) to be operated by a Payment Initiation Service Provider (PISP) TPP as defined by article 66 of [PSD2],
- Account Information Service (AIS) to be operated by an Account Information Service Provider (AISP) TPP as defined by article 67 of [PSD2], and
- Confirmation of the Availability of Funds service to be used by Payment Instrument Issuing Service Provider (PIISP) TPP as defined by article 65 of [PSD2].

For operating the new services a TPP needs to access the account of the PSU which is usually managed by another PSP called the Account Servicing Payment Service Provider (ASPSP). As shown in the following figure, an ASPSP has to provide an interface (called "PSD2 compliant Access to Account Interface" or short "XS2A Interface") to its systems to be used by a TPP for necessary accesses regulated by [PSD2]:



Further requirements on the implementation and usage of this interface are defined by a Regulatory Technical Standard (short RTS) from the European Banking Authority (short EBA), published in the Official Journal of the European Commission.

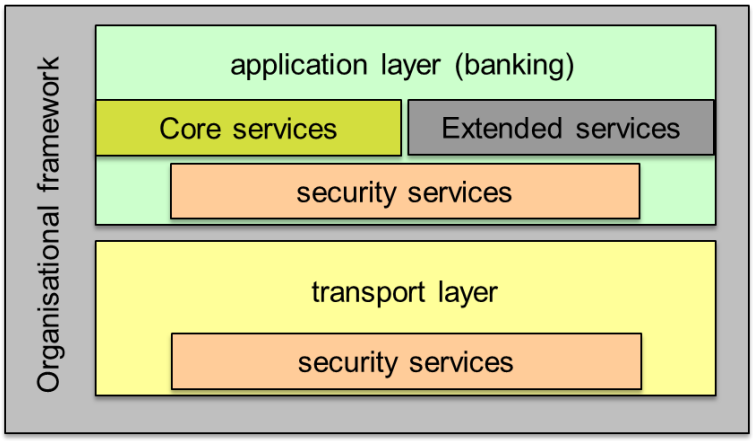
1.2 XS2A Interface Specification

This document is an extension of the NextGenPSD2 XS2A Specification which defines a standard for an XS2A Interface and by this reaching interoperability of the interfaces of ASPSPs at least for the core services defined by [PSD2].

The XS2A Interface is designed as a B2B interface between a TPP server and the ASPSP server. For time being, the protocol defined in this document is a pure client-server protocol, assuming the TPP server being the client, i.e. all API calls are initiated by the TPP. In future steps, this protocol might be extended to a server-server protocol, where also the ASPSP initiates API calls towards the TPP.

The Interoperability Framework defines operational rules, requirements on the data model and a process description in [XS2A-OR].

This document details the standard in defining messages and detailed data structures for **extended services** of the XS2A Interface. For the specification the two layers shown in the following figure are distinguished:



This document now first covers the extended service to give consent to the Confirmation of Availability of Funds service (short Confirmation of Funds service) through the XS2A Interface. This registration is managed by extending the consent mechanism of the core XS2A interface by a specific consent sub endpoint. After having the submitted consent authorised by the PSU with a PSU authentication towards the ASPSP, the TPP is registered for the related PSU account for confirmation of funds services.

The actual confirmation of funds service is defined in [XS2A-IG]. This service then can be used by the TPP. It is planned to introduce the consent-id as a conditional parameter to the confirmation of funds service as an erratum: If the consent of the PSU has been provided

through the consent mechanism described in this document, then the related consent-Id shall be delivered in related confirmation of funds services.

1.3 Structure of the Document

This document first outlines notations in Section 2 and requirements on the transport layer in Section 3. In Section 4, guiding principles for the definition of the provided extended services for the XS2A interface and the API structure with API endpoints and permitted access methods are described. Section 5 then specifies in detail how a consent of the PSU is established to enable the TPP to use an account of the PSU for a confirmation of funds service.

Remark for Future: Please note that the Berlin Group NextGenPSD2 XS2A interface is still under constant development. Technical issues, which are already in discussion within the Berlin Group NextGenPSD2 working structure are mentioned in this document by "Remark for Future" to make the reader aware of upcoming potential changes.

1.4 Document History

Version	Change/Note	Approved
Version 2.0		01 March 2019

2 Character Sets and Notations

For definition on character Sets and Notations as well as for request and response notations refer to Chapter 2 of [XS2A-IG].

3 Transport Layer

For details on the transport Layer, please refer to Chapter 3 in [XS2A-IG].

4 Application Layer: Guiding Principles

4.1 Signing Messages at Application Layer

The ASPSP may require the TPP to sign request messages. This requirement shall be stated in the ASPSP documentation. The signing requirements are defined in [XS2A-IG]. No specific requirements are defined for the Establish Confirmation of Funds Service.

4.2 API Access Methods

The following tables gives an overview on the HTTP access methods supported by the API endpoints and by resources created through this API for the extended services defined in this document.

Conditions in the following tables

It is further defined, whether this method support is mandated for the ASPSP by this specification or whether it is an optional feature for the ASPSP. Please note that this condition is given relative to the parent node of the path, i.e. the condition e.g. on a method on /v2/consents/confirmations-of-funds/{consentId} applies only if the endpoint /v2/consents/confirmation-of-funds is supported at all.

Please note that all methods submitted by a TPP, which are addressing dynamically created resources in this API, may only apply to resources which have been created by the same TPP before.

Examples

Please further note, that sections are referred in the Description's column. These sections provide examples for all related access methods.

4.2.1 Consents Endpoint

NOTE: In difference to version 1.x of the NextGenPSD2 XS2A Framework, the /consents endpoint is deeper structured in version 2.x of the NextGenPSD2 XS2A Framework as defined by the following table. The structure of the API, starting with the {consent-id} subendpoint itself stays unchanged. Hence, references to [XS2A-IG] made in the following table are to be applied to this substructure in direct analogy.

Endpoints/Resources	Method	Condition	Description
consents/confirmation-of-funds	POST	Optional	Create a consent resource to register a TPP for the confirmation of funds service for a given account by a PSU for a given PSU ID.

Endpoints/Resources	Method	Condition	Description
			See Section 5.4.1
consents/confirmation-of-funds/{consentId}	GET	Mandatory	Reads the exact definition of the given consent resource {consentId} including the validity status. Section 5.4.3
	DELETE	Mandatory	Terminate the addressed consent. Section 5.5
consents/confirmation-of-funds/{consentId}/status	GET	Mandatory	Read the consent status of the addressed consent resource. Section 5.4.2
consents/confirmation-of-funds/{consentId}/authorisations	POST	Mandatory	Create an authorisation sub-resource and start the authorisation process, might in addition transmit authentication and authorisation related data. The ASPSP might make the usage of this access method unnecessary, since the related authorisation resource will be automatically created by the ASPSP after the submission of the consent data with the first POST consents call. Cp. [XS2A-IG]
consents/confirmation-of-funds/{consentId}/authorisations/{authorisationId}	PUT	Mandatory for Embedded SCA Approach, Conditional for other approaches	Update data on the authorisation resource if needed. It may authorise 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.

Endpoints/Resources	Method	Condition	Description
			Cp. [XS2A-IG]
consents/confirmation-of-funds/{consentId}/authorisations/{authorisationId}	GET	Mandatory	Read the SCA status of the authorisation. Cp. [XS2A-IG]

4.3 Additional Error Information

No specific addition error information is needed for this extended service.

4.4 Status Information

4.4.1 Status Information for the AIS within the Establish Consent Process

No specific status information needed for this specific consent.

5 Consent to the Confirmation of Funds Service

Supported Sub-Services

This specification foresees only the registration of a TPP for the confirmation of funds service related to a payment account, which is authorised by a PSU through a PSU authentication with SCA.

Establishing Consent

The overall confirmation of funds service is separated in two phases:

- Establish a consent on a payment account between PSU and ASPSP to have dedicated TPPs using this dedicated account for Confirmation of Funds Requests.
- Performing the actual Confirmation of Funds Request as such.

The first part is not mandated to be offered by the XS2A interface following Article 65 (b), [PSD2], since this could also be an online banking function or even a paper based consent process. The extended service defined in this document offers this consent process as an XS2A functionality:

- Establish Confirmation of Funds Consent

  Within the service, the PSU is giving the consent to the ASPSP to grant a TPP (PIISP)

  - access to a dedicated account for the confirmation of funds service.

  This consent is then authorised by the PSU towards the ASPSP with an SCA.

  The result of this process is a consent resource. A link to this resource is returned to the TPP within this process. The TPP can retrieve the consent object by submitting a GET method on this resource. This object contains a.o. the detailed account information, the current validity and a Consent-ID token.

Using the Established Consent

This consent then can be referred to by the Confirmation of Funds Service implicitly or explicitly:

- The funds-confirmations endpoints implemented following [XS2A-IG] (version 1.3) is not using the related consent-ID. The related consent of the PSU is checked in the ASPSP's backend.
- The funds-confirmations endpoints implemented following future versions of the NextGenPSD2 XS2A CORE Interface are planned to support the related consent-

ID as HTTP header field in Confirmation of Funds Request, where the PSU consent has been managed through the service defined in this document.

5.1 Establish Consent for Confirmation of Funds Flow

The flows for the Establish Consent for Confirmation of Funds service are exactly the same as for other Establish Access to Account Information services, cp. [XS2A-IG].

5.2 Data Overview Establish Confirmation of Funds Consent Service

The following table defines the technical description of the abstract data model as defined [XS2A OR] for the account information service. The columns give an overview on the API protocols as follows:

- The "Data element" column is using the abstract data elements following [XS2A OR] to deliver the connection to rules and role definitions in this document.
- The "Attribute encoding" is giving the actual encoding definition within the XS2A API as defined in this document.
- The "Location" columns define, where the corresponding data elements are transported as HTTP parameters, resp. are taken from eIDAS certificates.
- The "Usage" column gives an overview on the usage of data elements in the different API Calls. Within [XS2A-OR], the XS2A calls are described as abstract API calls. These calls will be technically realised as HTTP POST, PUT, DELETE and GET commands. The calls are divided into the following calls:
    - Establish Consent Request, which shall be the first API Call for every transaction within XS2A Account Information service.
    - The Update Data Call is a call, where the TPP needs to add PSU related data, which is requested in the return of the first call. This call might be repeated.
    - The Authorisation Request is only used in an Embedded SCA Approach to authorise the transaction in case of a second factor is needed.
    - The Read Data Request is the request to retrieve Account Information data, which is addressed to different endpoints with different parameters.
    - The Status Request is used in cases, where the SCA control is taken over by the ASPSP and the TPP needs later information about the outcome.

The following usage of abbreviations in the Location and Usage columns is defined, cp. also [XS2A-OR] for details.

- x: This data element is transported on the corresponding level.
- m: Mandatory
- o: Optional for the TPP to use
- c: Conditional. The Condition is described in the API Calls, condition defined by the ASPSP

The following table does not only define requirements on request messages but also requirements on data elements for the response messages. **These requirements for data elements transported in the response body only apply in case of HTTP response code 2xx. In case of HTTP response code 4xx or 5xx requirements as defined in Sections on error processing contained in** [XS2A-IG] **apply.** In case of the Establish Consent Response Message, where a consent resource has only been created in case of a 2xx response code, e.g. no resource related information can be returned if the HTTP response code equals 4xx or 5xx.

Remark: The more technical functions like GET …/{consentId} and GET …/{authorisationId} and the Cancellation Request are not covered by this table.

Data element	**Attribute encoding**	**Location**					**Usage**							
		Path	Query Param.	Header	Body	Certificate	Establ.. Cons.Req.	Establ. Cons. Resp.	Upd. Data Req.	Upd. Data Resp	Auth. Req.	Auth Resp.	Status Req.	Status Resp.
Provider Identification		x					m		m		m		m	
TPP Registration Number						x	m		m		m		m	
TPP Name						x	m		m		m		m	
TPP Role						x	m		m		m		m	
TPP National Competent Authority						x	m		m		m		m	
Request Identification	X-Request-ID			x			m	m	m	m	m	m	m	m
Resource ID	consentId				x			m						
Resource ID[2]		x							m		m		m	
Access Token (from	Authorization			x			c		c		c		c	

[2] Please note that the Resource ID is transported in the path after the generation of the consent resource. This is then a path parameter without an explicit encoding of the attribute name.

Data element	**Attribute encoding**	**Location**					**Usage**							
		Path	Query Param.	Header	Body	Certificate	Establ.. Cons.Req.	Establ. Cons. Resp.	Upd. Data Req.	Upd. Data Resp	Auth. Req.	Auth Resp.	Status Req.	Status Resp.
optional OAuth2)														
TPP Signing Certificate Data	TPP-Signature-Certificate			x			c		c		c		c	
TPP Signing Electronic Signature	Signature			x			c		c		c		c	
Further signature related data	Digest			x			c		c		c		c	
ASPSP-SCA-Approach	ASPSP-SCA-Approach			x				c		c				
Transaction Status	consentStatus				x			m		m		m		m
SCA Status	scaStatus				x									o
PSU Message Information	psuMessage				x			o		o		o		o
TPP Message Information	tppMessages				x			o		o		o		o
PSU Identification	PSU-ID			x			c		c					
PSU Identification Type	PSU-ID-Type			x			c		c					
Corporate Identification	PSU-Corporate-ID			x			c		c		c		c	
Corporate Type	PSU-Corporate-ID-Type						c		c		c		c	
PSU Password	psuData.password				x				c					
Available SCA Methods	scaMethods				x			c		c				
Chosen SCA Method	chosenScaMethod				x				c					

Data element	Attribute encoding	Location					Usage							
		Path	Query Param.	Header	Body	Certificate	Establ.. Cons.Req.	Establ. Cons. Resp.	Upd. Data Req.	Upd. Data Resp	Auth. Req.	Auth Resp.	Status Req.	Status Resp.
PSU Authentication Data	psuData.authentication				x						m			
SCA Challenge Data	challengeData				x			c		c				
IP Address PSU	PSU-IP-Address			x			m		o		o		o	
PSU IP Port	PSU-IP-Port			x			o		o		o		o	
Further PSU related Information	PSU-Accept			x			o		o		o		o	
	PSU-Accept-Charset			x			o		o		o		o	
	PSU-Accept-Encoding			x			o		o		o		o	
	PSU-Accept-Language			x			o		o		o		o	
	PSU-Http-Method			x			o		o		o		o	
	PSU-Device-ID			x			o		o		o		o	
PSU User Agent	PSU-User-Agent			x			o		o		o		o	
GEO Information	PSU-Geo-Location			x			o		o		o		o	
Redirect URL ASPSP	_links.scaRedirect				x			c						
Redirect Preference	TPP-Redirect-Preferred			x			o							
Redirect URL TPP	TPP-Redirect-URI			x			c							
Authorisation Preference	TPP-Explicit-Authorisation-Preferred			x			o							
PSU Account	account				x		m							
Card Number	cardNumber				x		o							

Data element	Attribute encoding	Location					Usage							
		Path	Query Param.	Header	Body	Certificate	Establ.. Cons.Req.	Establ. Cons. Resp.	Upd. Data Req.	Upd. Data Resp	Auth. Req.	Auth Resp.	Status Req.	Status Resp.
Expiry Date	cardExpiryDate				x		o							
Card Information	cardInformation				x		o							
Registration Information	registrationInformation				x		o							

The XS2A Interface calls which represent the messages defined in [XS2A-OR] for the Establish Consent Request will be defined in the following sections.

PSU IP Address/Port and Further PSU related Information

The following data elements from the table above are forwarding information about the PSU-TPP interface and are enhancing the risk management procedures of the ASPSP. It is recommended to send these data elements in all request messages within the Establish Consent flow. The further definitions of request parameters within this section are not repeating the definition of these elements for the matter of better readability.

Attribute	Format	Condition	Description
PSU-IP-Address	String	Optional	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP.
PSU-IP-Port	String	Optional	The forwarded IP Port header field consists of the corresponding HTTP request IP Port field between PSU and TPP, if available.
PSU-Accept	String	Optional	The forwarded IP Accept header fields consist of the corresponding HTTP request Accept header fields between PSU and TPP, if available.
PSU-Accept-Charset	String	Optional	see above

Attribute	Format	Condition	Description
PSU-Accept-Encoding	String	Optional	see above
PSU-Accept-Language	String	Optional	see above
PSU-User-Agent	String	Optional	The forwarded Agent header field of the HTTP request between PSU and TPP, if available.
PSU-Http-Method	String	Optional	HTTP method used at the PSU – TPP interface, if available. Valid values are: • GET • POST • PUT • PATCH • DELETE
PSU-Device-ID	String	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.
PSU-Geo-Location	Geo Location	Optional	The forwarded Geo Location of the corresponding HTTP request between PSU and TPP if available.

5.3 Multi-currency Account Specifics for Confirmation of Funds Consent

The consent data model provides an account, where the consent for a Confirmation of Funds Request is granted on. The account currency is an optional sub field in the account reference.

In case of multi-currency accounts this implies that the default sub-account is addressed if no currency is submitted in the Establish Confirmation of Funds Request by the TPP. The default sub-account is set by the ASPSP.

5.4 Establish Confirmation of Funds Consent

In this section, the Establish Confirmation of Funds Consent process is defined for the XS2A Interface.

5.4.1 Confirmation of Funds Consent Request

Call

```
POST /v2/consents/confirmation-of-funds
```

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

No Side Effects

In difference to the Establish Account Information Consent as defined in [XS2A-IG], there is no side effect by the Establish Confirmation of Funds Consent Request.

Query Parameters

No specific query parameter.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
PSU-ID	String	Conditional	Might be mandated in the ASPSP's documentation, if OAuth is not chosen as Pre-Step.
PSU-ID-Type	String	Conditional	Type of the PSU-ID, needed in scenarios where PSUs have several PSU-IDs as access possibility.
PSU-Corporate-ID	String	Conditional	Might be mandated in the ASPSP's documentation. Only used in a corporate context.
PSU-Corporate-ID-Type	String	Conditional	Might be mandated in the ASPSPs documentation. Only used in a corporate context.
Authorization	String	Conditional	If OAuth2 has been chosen as pre-step

Attribute	Type	Condition	Description
			to authenticate the PSU.
TPP-Redirect-Preferred	Boolean	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.
TPP-Redirect-URI	String	Conditional	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. It is required that the domain of this URI is the same domain as contained in and secured by the web site certificate of the TPP.[3] **Remark for Future**: This field might be changed to mandatory in the next version of the specification.

[3] The term "same domain" in this specification means, that all labels needs to be identical except the left most label except label count would decrease below 2 (e.g. from "host1.subdomain.tld" a host in "same domain" might be "host2.subdomain.tld", where "tld" stands for top level domain.)

Attribute	Type	Condition	Description
TPP-Nok-Redirect-URI	String	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. It is required that the domain of this URI is the same domain as contained in and secured by the web site certificate of the TPP.
TPP-Explicit-Authorisation-Preferred	Boolean	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.

Request Body

Attribute	Type	Condition	Description
account	Account Reference	Mandatory	Account, where the confirmation of funds service is aimed to be submitted to.
cardNumber	Max35Text	Optional	Card Number of the card issued by the PIISP. Should be delivered if available.
cardExpiryDate	ISODate	Optional	Expiry date of the card issued by the PIISP
cardInformation	Max140Text	Optional	Additional explanation for the card product.

Attribute	Type	Condition	Description
registrationInformation	Max140Text	Optional	Additional information about the registration process for the PSU, e.g. a reference to the TPP / PSU contract

Response Code

HTTP Response Code equals 201.

Response Header

Attribute	Type	Condition	Description
Location	String	Mandatory	Location of the created resource.
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
ASPSP-SCA-Approach	String	Conditional	Possible values are: • EMBEDDED • DECOUPLED • REDIRECT OAuth will be subsumed by the constant value REDIRECT

Response Body

Attribute	Type	Condition	Description
consentStatus	Consent Status	Mandatory	authentication status of the consent
consentId	String	Mandatory	Identification of the consent resource as it is used in the API structure

Attribute	Type	Condition	Description
scaMethods	Array of Authenticati on Objects	Conditional	This data element might be contained, if SCA is required and if the PSU has a choice between different authentication methods. Depending on the risk management of the ASPSP this choice might be offered before or after the PSU has been identified with the first relevant factor, or if an access token is transported. If this data element is contained, then there is also an hyperlink of type "selectAuthenticationMethods" contained in the response body. These methods shall be presented towards the PSU for selection by the TPP.
chosenSca Method	Authenticati on Object	Conditional	This data element is only contained in the response if the ASPSP has chosen the Embedded SCA Approach, if the PSU is already identified with the first relevant factor or alternatively an access token, if SCA is required and if the authentication method is implicitly selected.
challengeData	Challenge	Conditional	It is contained in addition to the data element chosenScaMethod if challenge data is needed for SCA. In rare cases this attribute is also used in the context of the startAuthorisationWithPsuAuthentication or startAuthorisationWithEncryptedPsuAuthentica tion link.
_links	Links	Mandatory	A list of hyperlinks to be recognised by the TPP. Type of links admitted in this response (which might be extended by single ASPSPs as indicated in its XS2A documentation): "scaRedirect": In case of an SCA Redirect Approach, the ASPSP is transmitting the link to which to redirect the PSU browser. "scaOAuth": In case of an OAuth2 based

Attribute	Type	Condition	Description
			Redirect Approach, the ASPSP is transmitting the link where the configuration of the OAuth2 Server is defined. The configuration follows the OAuth 2.0 Authorisation Server Metadata specification. "startAuthorisation": In case, where an explicit start of the transaction authorisation is needed, but no more data needs to be updated (no authentication method to be selected, no PSU identification nor PSU authentication data to be uploaded). "startAuthorisationWithPsuIdentification": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the PSU identification data. "startAuthorisationWithPsuAuthentication": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the PSU authentication data. "startAuthorisationWithEncryptedPsu Authentication": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the encrypted PSU authentication data. "startAuthorisationWithAuthentication MethodSelection": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while selecting the authentication method. This link is contained under exactly the same conditions as the data element

Attribute	Type	Condition	Description
			"scaMethods" "startAuthorisationWithTransactionAuthorisation": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while authorising the transaction e.g. by uploading an OTP received by SMS. "self": The link to the Establish Account Information Consent resource created by this request. This link can be used to retrieve the resource data. "status": The link to retrieve the transaction status of the payment initiation. "scaStatus": The link to retrieve the scaStatus of the corresponding authorisation sub-resource. This link is only contained, if an authorisation sub-resource has been already created.
psuMessage	Max512Text	Optional	Text to be displayed to the PSU, e.g. in a Decoupled SCA Approach

Example

Request

```
POST https://api.testbank.com/v2/consents/confirmation-of-funds
Content-Type:           application/json
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7756
PSU-IP-Address:         192.168.8.78
PSU-User-Agent:          Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0)
Gecko/20100101 Firefox/54.0
Date:                   Sun, 06 Aug 2017 15:05:37 GMT

{
  "account":
          { "iban": "DE40100100103307118608" },
  "cardNumber": "1234567891234",
```

```
  "cardExpiryDate": "2020-12-31",
  "cardInformation": "MyMerchant Loyalty Card",
  "registrationInformation": "Your contract Number 1234 with MyMerchant is
completed with the registration with your bank."

}
```

Response in case *of a redirect*

```
HTTP/1.x 201 Created
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:     REDIRECT
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Location:               "v2/consents/confirmation-of-funds/1234-wertiq-983"
Content-Type:           application/json

{
  "consentStatus": "received",
  "consentId": "1234-wertiq-983",
  "_links": {
    "scaRedirect": {"href": "https://www.testbank.com/authentication/1234-
wertiq-983"},
    "status": {"href": "/v2/consents/confirmation-of-funds/1234-wertiq-
983/status"},
    "scaStatus": {"href": "v2/consents/confirmation-of-funds/1234-wertiq-
983/
authorisations/123auth567"}
  }
}
```

Response in case *of a redirect with a dedicated start of the authorisation process*

```
HTTP/1.x 201 Created
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:     REDIRECT
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Location:               "v2/consents/confirmation-of-funds/1234-wertiq-983"
Content-Type:           application/json

{
  "consentStatus": "received",
  "consentId": "1234-wertiq-983",
  "_links": {
    "startAuthorisation": {"href": "v2/consents/confirmation-of-funds/1234-
wertiq-983/authorisations"}
  }
```

```
}
```

Response *in case of the OAuth2 approach with an implicit generated authorisation resource*

```
HTTP/1.x 201 Created
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:     REDIRECT
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Location:               "v2/consents/confirmation-of-funds/1234-wertiq-983"
Content-Type:           application/json

{
  "consentStatus": "received",
  "consentId": "1234-wertiq-983",
  "_links": {
    "self": {"href": "/v2/consents/confirmation-of-funds/1234-wertiq-983"},
    "scaStatus": {"href": "v2/consents/confirmation-of-funds/1234-wertiq-
983/authorisations/123auth567"},
    "scaOAuth": {"href": "https://www.testbank.com/oauth/.well-known/oauth-
authorization-server"}
  }
}
```

Response in case of the decoupled approach

```
HTTP/1.x 201 Created
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:     DECOUPLED
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Location:               "v2/consents/confirmation-of-funds/1234-wertiq-983"
Content-Type:           application/json

{
  "consentStatus": "received",
  "consentId": "1234-wertiq-983",
  "_links": {
    "startAuthorisationWithPsuIdentification": {"href":
"/v2/consents/confirmation-of-funds/1234-wertiq-983/authorisations"}
  }
}
```

Response in case of the embedded approach

```
HTTP/1.x 201 Created
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
ASPSP-SCA-Approach:     EMBEDDED
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Location:               "v2/consents/confirmation-of-funds/1234-wertiq-983"
Content-Type:           application/json

{
  "consentStatus": "received",
  "consentId": "1234-wertiq-983",
  "_links": {
    "startAuthorisationWithPsuAuthentication": {"href":
"/v2/consents/confirmation-of-funds/1234-wertiq-983/authorisations"}
  }
}
```

5.4.2 Get Status Request

Call

`GET /v2/consents/confirmation-of-funds/{consentId}/status`

Can check the status of an account information consent resource.

Path Parameters

Attribute	Type	Description
consentId	String	The consent identification assigned to the created resource.

Query Parameters

No specific query parameters defined.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Attribute	Type	Condition	Description
Authorization	String	Conditional	Is contained only, if an OAuth2 based SCA was performed in the corresponding consent transaction or if OAuth2 has been used in a pre-step.

Request Body

No request body.

Response Code

HTTP Response Code equals 200.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
consentStatus	Consent Status	Mandatory	This is the overall lifecycle status of the consent.

Example

Request

```
GET https://api.testbank.com/v2/consents/confirmation-of-
funds/qwer3456tzui7890/status
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
PSU-IP-Address:         192.168.8.78
PSU-User-Agent:          Mozilla/5.0 (Windows NT 10.0; WOW64; rv:54.0)
Gecko/20100101 Firefox/54.0
Date:                   Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{
  "consentStatus": "valid"
}
```

5.4.3 Get Consent Request

Call

```
GET /v2/consents/confirmation-of-funds/{consentId}
```

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.

Path Parameters

Attribute	Type	Description
consentId	String	ID of the corresponding consent object as returned by an Account Information Consent Request

Query Parameters

No specific query parameter.

Request Header

The same as defined in Section 5.4.2.

Request Body

No request body.

Response Code

HTTP Response Code equals 200.

Response Header

The same as defined in Section 5.4.2.

Response Body

Attribute	Type	Condition	Description
account	Account Reference	Mandatory	Account, where the confirmation of funds service is aimed to be submitted to.
cardNumber	Max35Text	Optional	Card Number of the card issued by the PIISP. Should be delivered if available.
cardExpiryDate	ISODate	Optional	Expiry date of the card issued by the PIISP
cardInformation	Max140Text	Optional	Additional explanation for the card product.
registrationInformation	Max140Text	Optional	Additional registration information.
consentStatus	Consent Status	Mandatory	The status of the consent resource.

Example

Request

```
GET https://api.testbank.com/v2/consents/confirmation-of-
funds/qwer3456tzui7890
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{
  "account":
          { "iban": "DE40100100103307118608" },
  "cardNumber": "1234567891234",
  "cardExpiryDate": "2020-12-31",
  "cardInformation": "MyMerchant Loyalty Card",
  "consentStatus": "valid"
}
```

5.4.4 Multilevel SCA for Establish Consent

The Establish Confirmation of Funds Consent Request messages defined in this section are independent from the need of one or several SCA processes, i.e. independent from the number of authorisations needed for establishing the consent. In contrast, the Establish Confirmation of Funds Consent Response messages defined above in this section are specific to the processing of one SCA. In the following the background is explained on diverging requirements on the Establish Confirmation of Funds Consent Response messages.

For establish confirmation of funds consent with multilevel SCA, this specification requires an explicit start of the authorisation, i.e. links directly associated with SCA processing like "scaRedirect" or "scaOAuth" cannot be contained in the response message of a Establish Confirmation of Funds Consent Request for a consent, where multiple authorisations are needed. Also if any data is needed for the next action, like selecting an SCA method is not supported in the response, since all starts of the multiple authorisations are fully equal. In these cases, first an authorisation sub-resource has to be generated following the "startAuthorisation" link.

Response Body for Establish Confirmation of Funds Messages with Multilevel SCA

Attribute	Type	Condition	Description
consentStatus	Consent Status	Mandatory	The values defined in [XS2A-IG] for consent resources might be used.
consentId	String	Mandatory	resource identification of the generated payment initiation resource.
_links	Links	Mandatory	"startAuthorisation": In case, where an explicit start of the transaction authorisation is needed, but no more data needs to be updated (no authentication method to be selected, no PSU identification nor PSU authentication data to be uploaded). "startAuthorisationWithPsuIdentification": The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the PSU identification data. "startAuthorisationWithPsuAuthentication": The link to the authorisation end-point, where an authorisation sub-resource has to be generated

Attribute	Type	Condition	Description
			while uploading the PSU authentication data. "startAuthorisationWithEncryptedPsuAuthentication": The link to the authorisation end-point, where an authorisation sub-resource has to be generated while uploading the encrypted PSU authentication data. "self": The link to the consent resource created by this request. This link can be used to retrieve the resource data. "status": The link to retrieve the status of the consent.
psuMessage	Max512Text	Optional	Text to be displayed to the PSU
tppMessages	Array of TPP Message Information	Optional	Messages to the TPP on operational issues.

Remark: In difference to the Establish Confirmation of Funds Consent Flow with one SCA, optimisation processes with implicitly generating authorisation sub-resources are not supported for Multiple SCA to keep the several authorisation processes of different PSUs for the same consent identical, so that the start of the authorisation process is context free. That is, the only steering hyperlinks returned to the TPP after starting establishing a consent are "start authorisation" hyperlinks with information in addition about mandatory data to be uploaded with the Start Authorisation Request (PSU Identification or PSU Authentication data). It is not possible to upload with the first command the selected authentication method or OTP Response data because this would require to transport the selected authentication methods or challenge data before.

5.5 Revoke a Confirmation of Funds Consent

The TPP can revoke an account information consent object if needed with the following call:

Call

```
DELETE /v2/consents/confirmation-of-funds/{consentId}
```

Deletes a given consent.

Path Parameters

Attribute	Type	Description
consentId	String	Contains the resource-ID of the consent to be deleted.

Query Parameters

No specific query parameters.

Response Code

The HTTP response code equals 204 for a successful cancellation.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained only, if an OAuth2 based SCA was performed in the corresponding consent transaction or if OAuth2 has been used in a pre-step.

Request Body

No Request Body.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

No Response Body

Example

Request

```
DELETE https://api.testbank.com/v2/consents/confirmation-of-
funds/qwer3456tzui7890
X-Request-ID            99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date                    Sun, 13 Aug 2017 17:05:37 GMT
```

Response

```
HTTP/1.x 204 No Content
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
```

6 Data Types and Codes specific to Extended Services

No specific data types or codes are needed for the Establish Confirmation of Funds Consent is required. All referred data types and codes are defined in [XS2A-IG].

7 References

[XS2A-OR] NextGenPSD2 XS2A Framework, Operational Rules, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface, version 1.0, published 08 February 2018

[XS2A-IG] NextGenPSD2 XS2A Framework, Implementation Guidelines, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface, version 1.3, published November 2018

[XS2A-DP] NextGenPSD2 XS2A Framework, Domestic Payment Definitions, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface, current version

[EBA-RTS] Commission Delegated Regulation (EU) 2018/389 of 27 November 2017 supplementing Directive 2015/2366 of the European Parliament and of the Council with regard to Regulatory Technical Standards for Strong Customer Authentication and Common and Secure Open Standards of Communication, C(2017) 7782 final, published 13 March 2018

[eIDAS] Regulation (EU) No 910/2014 of the European Parliament and of the Council on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market, 23 July 2014, published 28 August 2014

[PSD2] Directive (EU) 2015/2366 of the European Parliament and of the Council on payment services in the internal market, published 23 December 2015