Handbook

XS2A Redirect

Sparkassen-Finanzgruppe

finanz informatik

Contents

1 Revision History

Version	Date	Revision
1.0	08.28.2019	Initial publication
1.1	10.02.2019	Chapter 4: Correction attribute name clientID and reference to space after scope added. Chapter 7: Note on different spelling clientID/ client-id added.
1.2	29.10.2019	Chapter 4: Correction Scope-Coding
1.2.1	15.11.2019	Chapter 4:
1.2.2	13.10.2020	Chapter 4: Correction attribute names for Redirect-Parameter
1.3	02.06.2021	Chapter 4: Add attribute restrictions
1.4	12.01.2024	Correction in Chapter 5 and 7

2 Intro

With the September 2019 release, we support authorization using the redirect approach in combination with OAuth2. The bank-offered consent is also supported in this context whereby the user/PSU determines the authorization of the consent via our online banking system.

Subsequent access to authorized resources using the redirect approach must then be performed with a mandatory OAuth access token.

The authorization using the redirect approach takes place via the Internet presence (online banking) of the Sparkassen/financial institutions.

This documentation describes the use of our redirect solution. Basic information is provided in the Berlin Group Specification and RFC 6749/8414 and will not be described redundantly here.

3 Creating a Resource

When you create a resource (consent, payment, payment cancellation), the TPP-Redirect-Preferred header attribute determines whether to authorize using the redirect or the embedded approach. If you want to authorize the resource using the embedded approach, you must specify this explicitly (TPP-Redirect-Preferred=false). If the attribute is missing, we automatically assume that the authorization should take place using the redirect approach.

In the redirect approach the TPP-Redirect-URI header attribute is a mandatory field. However, we also recommend that you use TPP-Nok-Redirect-URI so that you can distinguish between success and error events in the sequence. To be able to assign the received authorization code to your process during a redirect, please use the "state" variable (see the following section as well). Because of this, it is not necessary to include custom parameters in one of the redirect URIs; OAuth2 framework parameters are used, which are specially designed for this application. You can nevertheless optionally specify additional parameters in the redirect URL. These are transmitted unchanged during a redirect.

You can change the approach when canceling payments. In other words, a payment authorized using the embedded approach can be canceled using the redirect approach and vice versa.

If you have created the resource, you will obtain a "scaOAuth2" link instead of the "startAuthorisationWithPsuAuthentication" link.

4 Redirecting the PSU

Important. Please note:
If you create Consents or Payments using the redirect mode the link “scaOAuth2” instead of “startAuthorisationWithPsuAuthentication” is returned.

This link is the link to the OAuth Well Known endpoint.

For the first step, you have to determine the authorization server's metadata using the OAuth-Well-Known endpoint (see "scaOAuth2" link). This endpoint provides RFC-8414-compliant URLs for the authorization endpoint (Sparkassen Internet presence/online banking) and for the endpoint in order to create tokens.

Example:

```
{
  "issuer": "https://xxx",
  "authorization_endpoint": "https://yyy ",
  "token_endpoint": "https://zzz"
}
```

In the second step, you must complete the OAuth parameters for the resource and redirect the user there.

Parameter	Description	Required/Optional
response_type	Indicates what kind of response the client expects. It is possible to request tokens directly. In this implementation, however, only authorization codes (=code) can be requested.	Required Fix: code
client_id	Your client ID of the TPP according to the Berlin Group Specification protocol. Specifically, this concerns the subjectNcaId from your certificate.	Required Max: 128
scope	Scope for which access is being requested (i.e., for a consent or payment). Allocated according to the Berlin Group Specification as follows: "PIS“ + Colon („%3A“) + <PaymentID>" or "AIS:“ + Colon („%3A“) + <ConsentID>"	Required Max: 40

state	Variable returned during a redirect that allows you to assign the authorization code you received to your original process.	Required Max: 255
code_challenge	Challenge per PKCE (Proof Key for Code Exchange RFC 7636)	Required Max: 128
code_challenge_method	Method per PKCE (RFC 7636)	Required Fix: S256

The following attributes are limited in the character set:

- client_id
- state
- code_challenge

Allowed: ("^[a-zA-Z0-9_-]+.*$")

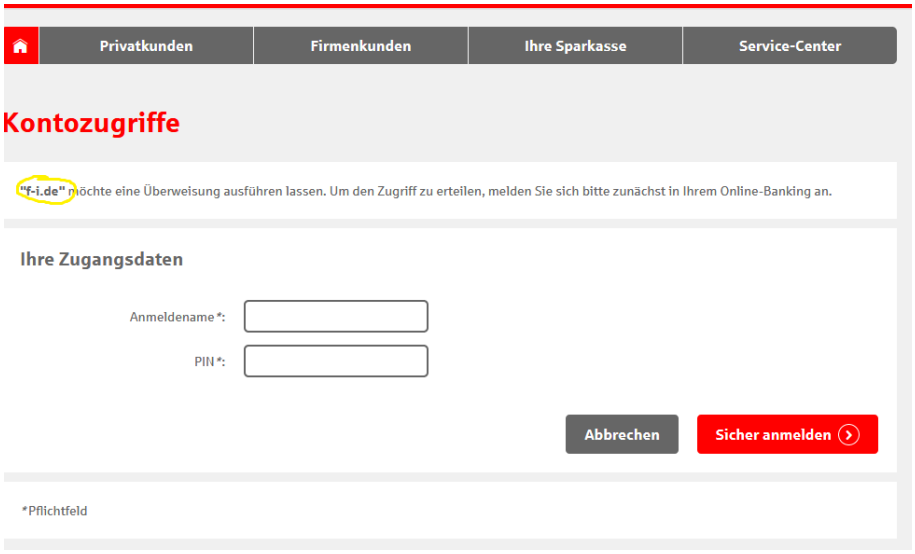
Illustrative example URL for redirecting a user:

```
https://www.sparkasse-xxx.de/.../xs2a/authorize?
        response_type=code&
        client_id=<TPP Registration number>&
        scope=AIS%3A<consentId>&
        state=S8NJ7uqk5fY4EjNvP&
        code_challenge_method=S256&
        code_challenge=vXVXiMA4CQ_Buik94dCNpfIfveWdNxMEwVtxGDz7xWg
```

5 Sparkassen Internet presence

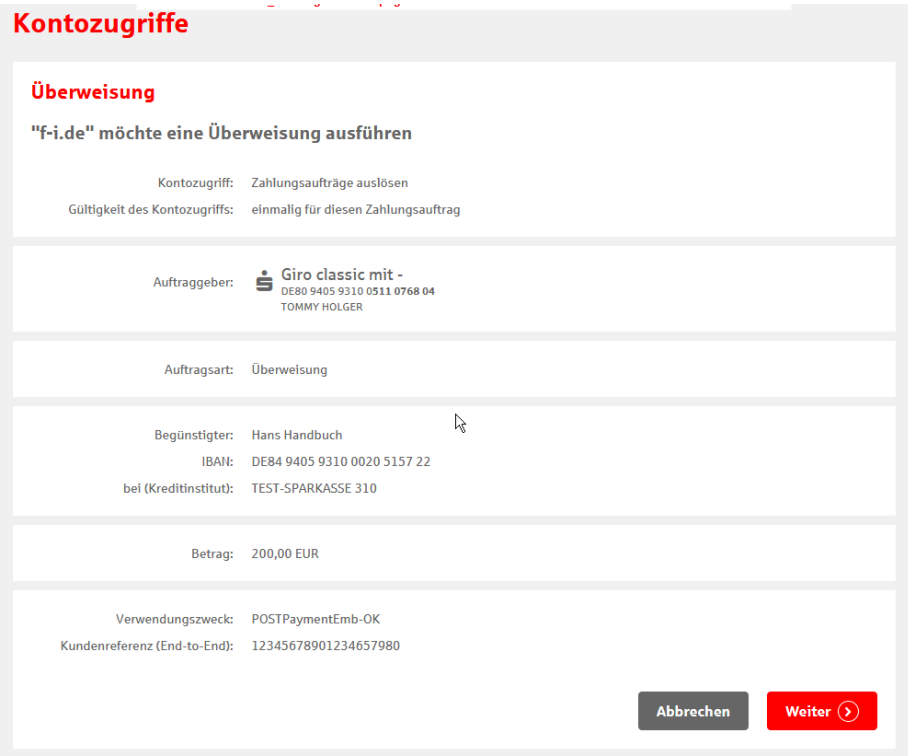
Even before logging in, the user receives information about which service provider would like to begin an authorization.

Example with a TPP named f-i.de:



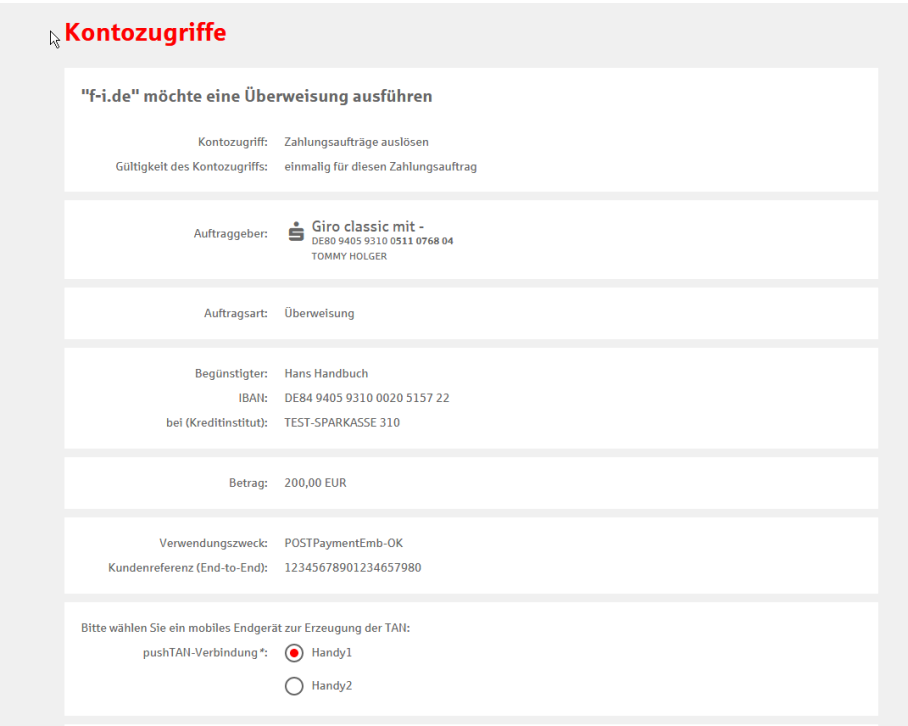
The user then logs on and receives detailed information about the resource (consent, payment, payment cancellation).

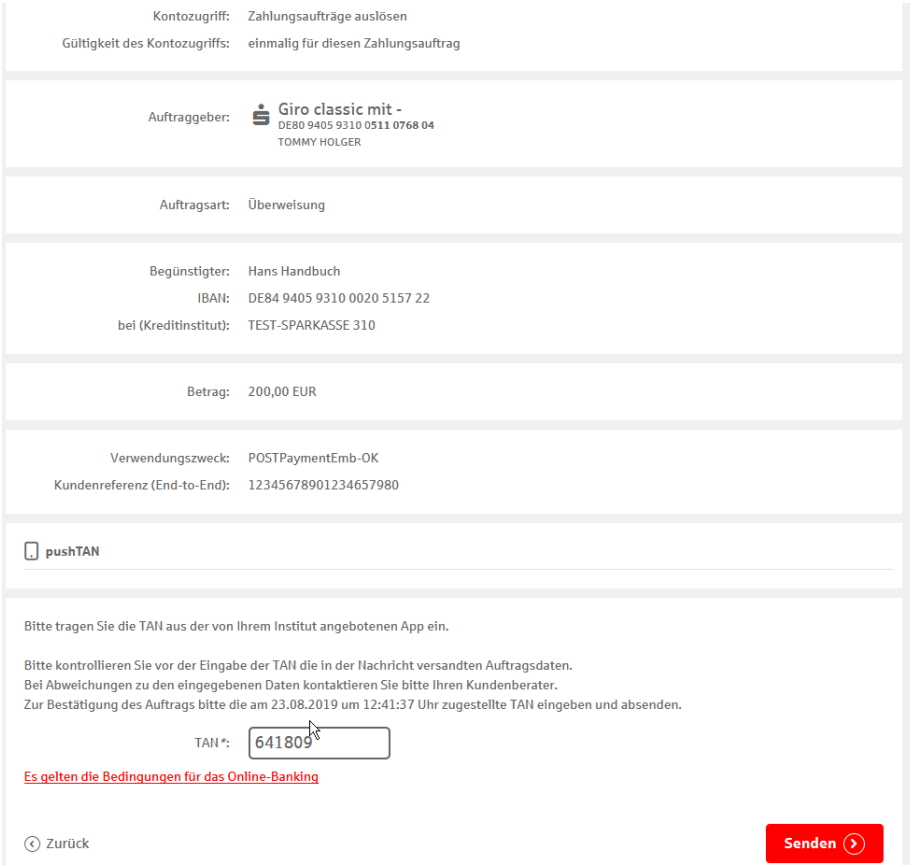
Example with a scheduled transfer:



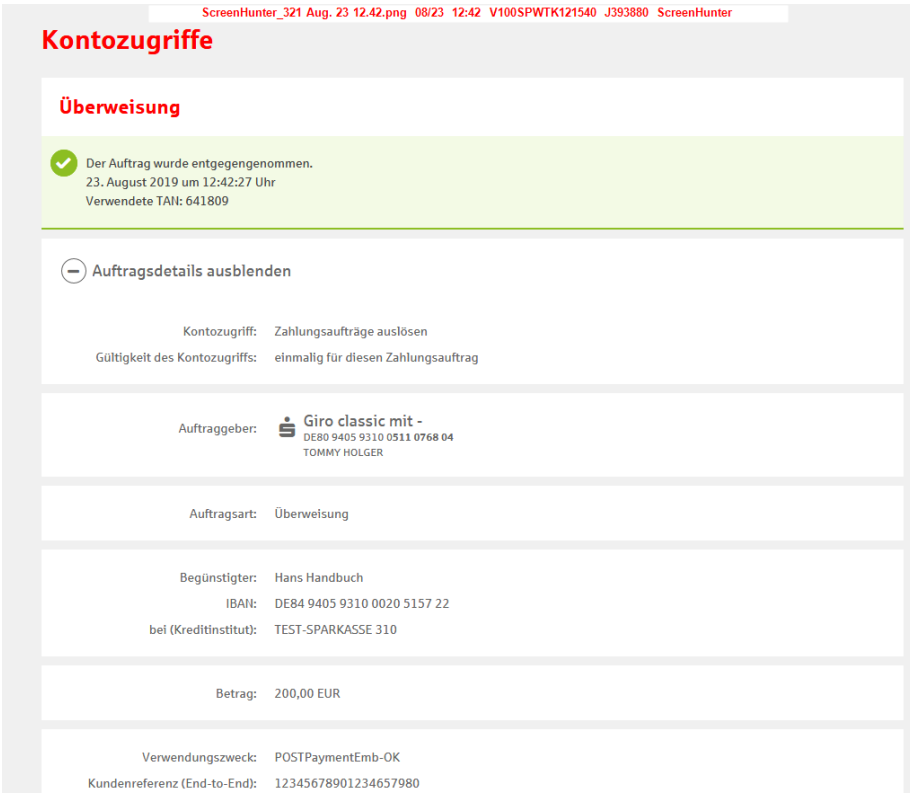
Subsequently, if several TAN methods are available, the user must then select the TAN method and enter the TAN. All TAN methods are supported here (pushTAN and chipTAN) at the same time as all other access channels.

Example selecting a pushTAN connection:





If successful, the user is asked to log off and then returned to the URI (TPP-Redirect-URI) you have saved. In the event of an error, we will redirect to the TPP-Nok-Redirect-URI as long as it has been saved or alternatively to the TPP-Redirect-URI.



Use of the Internet presence is limited to the authorization of the resource you have presented. This is to ensure that the return to your application indeed takes place.

Also, the version of the Internet presence that is designed for redirects does not support management transactions such as PIN changes, changing TAN methods and so forth.

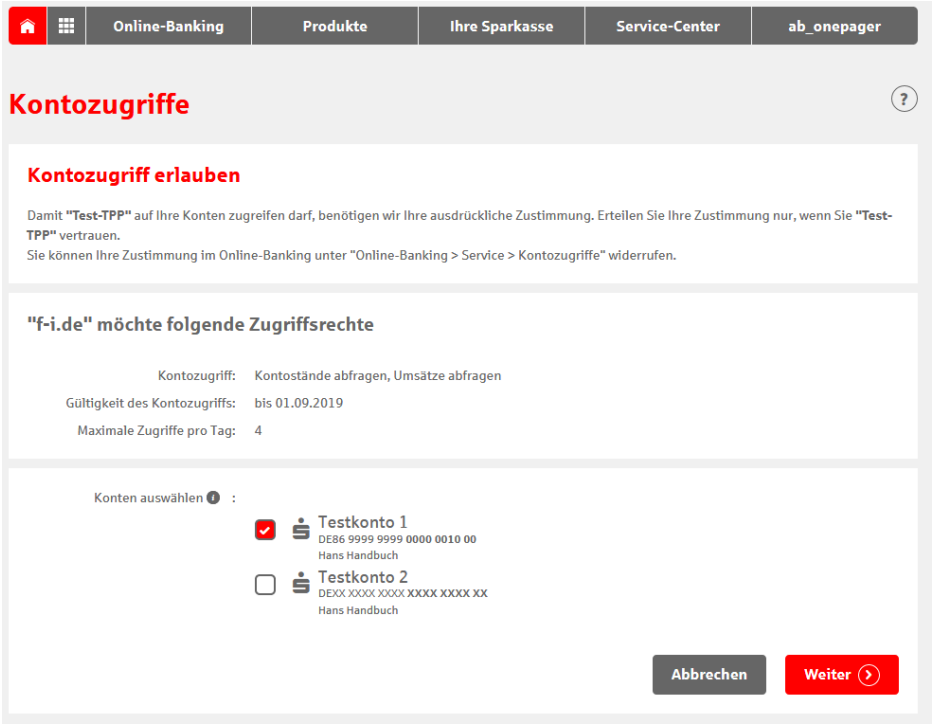
If the user wants to make use of other features, he or she must re-login to the standard Internet presence of his or her Sparkasse/financial institution.

5.1 Creating a bank-offered consent

With a bank-offered consent, the customer may stipulate the accounts to which he or she would like to grant the consent. This is done via the Internet presence.

In the sandbox, the selection by the customer is simulated. An account (DE86 9999 9999 0000 0010 00) is selected by default.

Example of account selection by the customer:



6 Redirecting with AuthCode

If the authorization has been completed successfully, you will receive an AuthCode: <TPP-Redirect-URI>?state=<state>&code=<authorization_code>. You must then use this to retrieve an access token. The procedure is time-sensitive (10 minutes). The access token has to be retrieved immediately in succession.

In the event of an error, no access token is provided.

Exception for a second signature (order confirmation): If a payment has to be authorized by two or more users, you will only receive an AuthCode when the entire authorization is complete.

7 Exchanging AuthCode for OAuth Token

We issue RFC-compliant access and refresh tokens. You can obtain the URI with the "token_endpoint" link using the OAuth-Well-Known endpoint.

Example body of a request:

'code=XJAy0ajuvQfv5FwqAi9lmP_LjllXDVtwbtUECAuTlIA&client_id=meinTpp&code_verifier=E6vTY2ZMFctVP16Ei7617SOeg1gexjIpELwIwOOiVE4&grant_type=authorization_code'

Note: In this endpoint, the attribute is called client_id, while in the URL it is called clientId. We will correct this promptly and then support both spellings.

An example response:


```
{
"access_token":
"eyJhbGciOiJSUzI1NiJ9.eyJzdWIiOiJQU0QyIiwiU2NvcGUiOiJQSVM6IDEyMzQtd2VydGlxLTk4MyIsImlzcyI6IkZpbmFueiBJbmZvcm1hdGlrIiwiZXhwIjoxNTY2NTUyMzI1fQ.JG0EGwIRgEfiq8mfdoocjEJzGEsMXfetDekSMLI6a4ku4jXdCDk-
sN663fNQ4UaXC1SEebs0xAm5IYvMLneedpPcBvAsNLf5vch4RjgeUymv1bYPlaVawXt7wsudhJJCT5iT0_L_Nf_Uwli2G5dcm_WMo4l0JSJgqhSX25gG-giaggLZ-
sury3GGC335zJVef8GtH9a42aYMba6KjQtdLbKGIz492hBBVswJe4khaURZ7QyiZ0RnI5huSKnesdHes9C1EL9UL6v3ue8et9yYrO7hIO9lrIc-
lacnKweAXCgD8r2EGkmUHsxgruLoD8JBDdtJ5JiUqdU-4W08rmqoAA",
  "token_type": "Bearer",
  "expires_in": "3600",
  "refresh_token": "mnllysEZDis-teXu6S4MItliLhvb8nnVlGwtKvXSc5M"
}
```


An access token that we have issued is currently valid for one hour (3,600 seconds). You can also derive the validity from the token.

In the initial version, the refresh token is valid for 180 days. In the future we will adjust the validity of the refresh token to match the validity of the resource.

Information about the code challenge and the verifier: Proof Key for Code Exchange (PKCE) support eliminates the risk of an attacker being able to exchange an intercepted authorization code for a token because it also requires the verifier to create tokens. Details on this are published under RFC 7636.

8 Using OAuth Tokens

The OAuth token is required only for accessing successfully authorized resources. This allows you to query the status of resources or authorizations even without an AuthCode or tokens.

The token is mandatory for the following endpoints:

- Read account list
- Read transaction list
- Read balance
- Read account details
- Payment cancellation request (if the payment in question has already been successfully authorized)

The token is optional for all other endpoints.

If a token is sent, then it must exist and be valid. The lifecycle of the access/refresh token is not synchronized with the lifecycle of the XS2A resource. In other words, the status of the resource is always verified, and the request is rejected despite the token being valid. Example: A user (PSU) revokes a consent.