

Sandbox příručka pro službu Iniciování platby

Change log

Date	Version	Description
26.02.2020	5	Aktualizace dokumentu a printscreenů.
21.05.2020	6	Rožšíření dokumentu o trvalé příkazy (platné od 27.5.2020)
03.06.2022	7	Rožšíření dokumentu o dávkové platby
20.08.2024	8	Aktualizace dokumentu (zejména printscreenů) z důvodu nového prostředí pro klienty (nový API store).

Obsah

1. Služba Iniclace platby (PIS)	4
2. Sandbox API pro volání služby Iniclace platby.....	4
3. Vydání certifikátu.....	4
4. Definice mocku.....	4
6. Postup pro vygenerování klíče/tokenu ke službě PISP	4
Vložení tokenu do PISP requestů	5
7. Postup přístupu do aplikace přes konzoli na API portálu	7
Volání mocku PIS „New Standing Order initiation“ k otestování.....	7
Volání mocku PIS „Standing Orders Authorization - start“ k otestování	10
Volání mocku PIS „Standing Order Status“ k otestování.....	13
Volání mocku PIS „Standing Order Detail“ k otestování.....	15
Volání mocku PIS „New batch payment - instruction initiation“ k otestování	17
Volání mocku PIS „Batch Authorization – starting the particular authorization method“ k otestování	19
Volání mocku PIS „Batch payment status“ k otestování	22
8. Postup přístupu do aplikace přes přímé volání	24
Nový trvalý příkaz – iniciace platby (POST/my/standingorders).....	24
Elementy zprávy Nová platba – iniciace trvalého příkazu	25
Elementy response Nový trvalý příkaz – iniciace trvalého příkazu	29
Iniclace autorizace trvalé platby – specifické pro každou banku (POST / standingorders/{transactionIdentification}/sign/{signId})	30
Elementy zprávy Iniclace autorizace trvalého příkazu.....	31
Parametry requestu:.....	31
Detail trvalého příkazu (GET /my/standingorders/{transactionIdentification}).....	32
Elementy zprávy Detail trvalého příkazu	33
Elementy response zprávy Detail trvalého příkazu	33
Status založeného/iniciovaného trvalého příkazu (GET /my/standingorders/{transactionIdentification})/status	37
Elementy zprávy Status založeného/iniciovaného trvalého příkazu	38
Dávková platba – iniciace (POST /my/batchpayments).....	39
Elementy request Dávková platba – iniciace	39
Elementy response Dávková platba – iniciace	40
Chybové kódy definované pro službu POST iniciace dávkové platby	41
Elementy request iniciace autorizace dávkové platby	42
Elementy response iniciace autorizace dávkové platby	43
Chybové kódy definované pro službu POST iniciace autorizace dávkové platby.....	43
Dávková platba – status (GET/my/batchpayments/{transactionIdentification}/status)	45
Elementy response Dávková platba – Status.....	45
Chybové kódy definované pro službu GET status dávkové platby	46

1. Služba Iniciační platby (PIS)

Přehled implementovaných zdrojů:

- Trvalý příkaz – iniciační platby
- Iniciační autorizovaného trvalého příkazu
- Status a detail trvalého příkazu
- Dávková platba – iniciační
- Iniciační autorizace dávkové platby
- Status dávkové platby

Nepovolené typy plateb pro prostředí Sandbox:

- Inkasa (povolení k inkasu, příkaz k inkasu)
- Instantní platby
- Platba šekem

Komerční banka vychází z jednotné struktury a formátu informací, která byla vydefinována Českou bankovní asociací v dokumentu [Czech Open Banking Standard](#).

Informace poskytované prostřednictvím Otevřeného bankovní API jsou lokalizovány do českého i anglického jazyka.

Povolená znaková sada vychází ze znakové sady SWIFT pro zahraniční/SEPA platby.

Během jednoho volání může být zaslán a zpracován vždy jen jeden dotaz.

2. Sandbox API pro volání služby Iniciační platby

Sandbox umožňuje třetím stranám vyzkoušení služby (mock) pro získávání níže uvedených informací, vztahujících se k platebnímu účtu klienta Komerční banky, a.s. (dále jen Komerční banky).

Do sandboxu API mohou přistupovat jakékoliv subjekty, tedy ne jen třetí strany s licenci pro PSD2 služby. Nutnou podmínkou je registrace na API portálu KB <https://developers.kb.cz/>. Bez registrace není možné služby na sandboxu využívat. Postup registrace je popsán v dokumentu API Sandbox Registrace.

3. Vydání certifikátu

Certifikát je nezbytnou podmínkou jak pro produkční volání, tak pro sandbox PSD2. Pro využití na sandboxu budou třetím stranám po jejich registraci na Sandboxu poskytnuty certifikáty od Komerční banky **na základě žádosti zasláné na emailovou schránku api@kb.cz**. Certifikáty ze sandboxu nejsou určeny pro produkční využití. Na produkci jsou taková volání zamítána a monitorována. Na produkci se používají kvalifikované PSD2 certifikáty vydané kvalifikovanou certifikační autoritou dle EU QTSP seznamu na <https://webgate.ec.europa.eu/tl-browser/>

4. Definice mocku

Jsou definovány vzorové requesty pro statické volání, které není možné ve volání třetí stranou měnit. V případě změny bude request zamítnut. Dle použitých vzorových requestů jsou vráceny odpovídající response, případně error kódy.

5. Hlášení chyb

Hlášení chyb Sandboxu nebo jeho volání probíhá vždy pomocí emailové schránky api@kb.cz. Odeslaný email musí obsahovat níže uvedené náležitosti. V případě chybějící požadované informace nebude možné dotaz nebo chybu zpracovat.

PSD2 API: CZ, SK

Prostředí: Sandbox, Produkce

Zda bylo voláno z FE Sandbox vč. typu a verze použitého prohlížeče nebo v případě BE volání název a verzi programu pro BE volání

Typ volání

Datum a čas uskutečnění volání

IP adresu

Chybu a její co nejpresnější popis, který může být doplněn o příslušný otisk obrazovky

Bez výše uvedených hodnot se nebude možné hlášenou chybou zabývat.

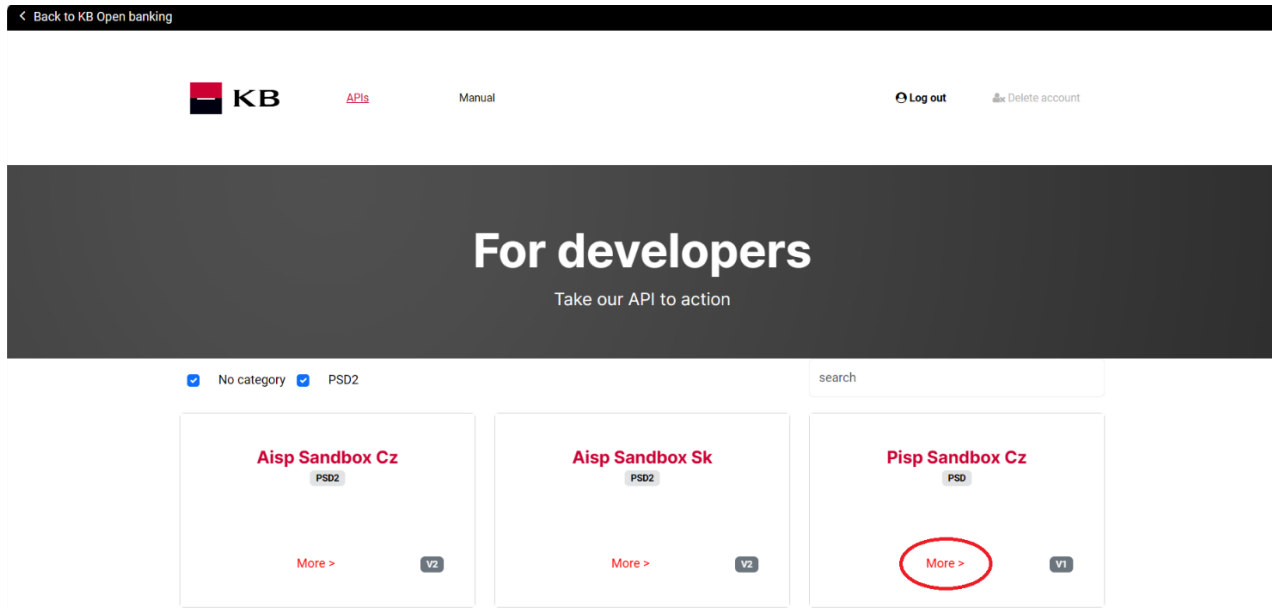
6. Postup pro vygenerování klíče/tokenu ke službě PISP

Předpoklady pro vygenerování klíče/tokenu

Uživatel je řádně zaregistrován a přihlášen do portálu Sandbox.

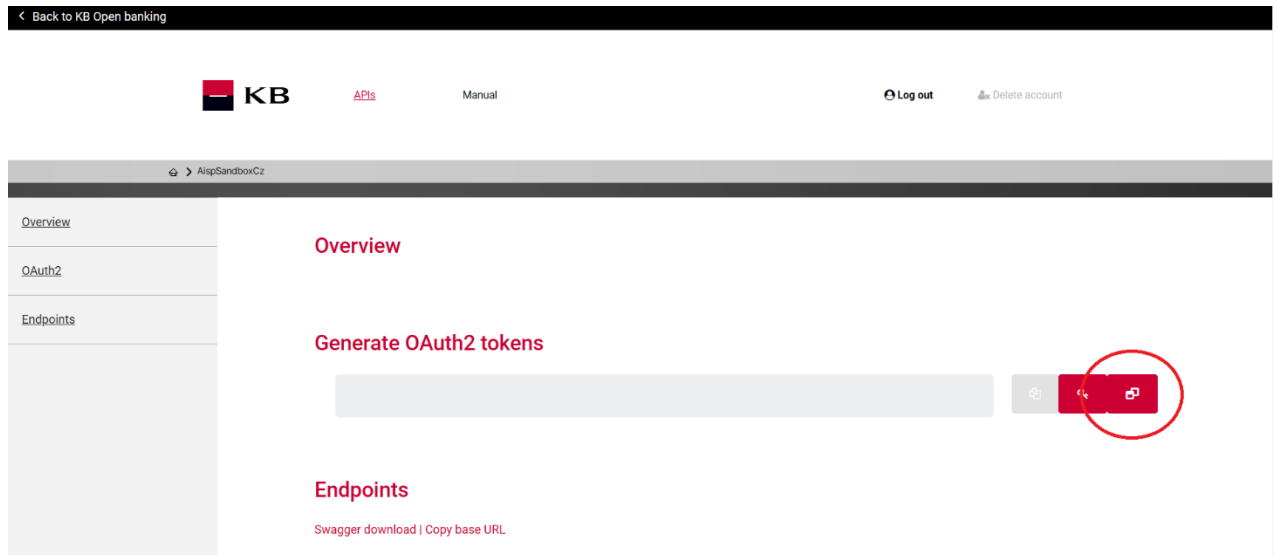
Vstup do PISP API

Přihlášený uživatel vstoupí do PISP API kliknutím na nápis „More“ na dlaždičce s PISP API.



Vstup do aplikačního menu a výběr požadované aplikace

Do aplikačního menu vstoupí pomocí odkazu Applications.



Generování certifikátu pro službu PISP

Uživatel v Sandboxu může zvolit vygenerování tokenu pro službu PISP za předpokladu, že je uživatel přihlášen k odběru služeb PISP. Postup generování tokenu je popsán v příručce Sandbox OAuth.

Vložení tokenu do PISP requestů

Uživatel klikne na tlačítko „Authorize“. Zobrazí se mu dialogové okno, kde vyplní token a poté jej uloží

kliknutím na tlačítko „Authorize“.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers

Authorize

PISP

POST	/standingorders	New standing order initiation	createNewStandingOrder		
POST	/standingorders/{transactionIdentification}/sign/{signId}	Request for authorization of standing order	authorizeStandingOrder		
GET	/standingorders/{transactionIdentification}/status	Standing orders status	getStandingOrderStatus		
GET	/standingorders/{transactionIdentification}	Standing orders detail	getStandingOrderById		
POST	/batchpayments	New batch payment - instruction initiation	batchPaymentInitiation		
POST	/batchpayments/{transactionIdentification}/sign/{signId}	Batch Authorization – starting the particular authorization method	batchAuthorizationWithCode		
GET	/batchpayments/{transactionIdentification}/status	Batch payment status	batchPaymentsStatus		

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP-Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers

Authorize

PISP

POST	/my/payments	New payment - instruction initiation	paymentCreate		
POST	/my/payments/{paymentId}/sign/{signId}	Payment Authorization – starting the particular authorization method	paymentAuthorizationWithCode		
GET	/my/payments/{paymentId}	Payment detail information	paymentDetail		
DELETE	/my/payments/{paymentId}	Delete unauthorized payment	paymentDelete		
GET	/my/payments/{paymentId}/status	Payment status information	paymentStatus		

Available authorizations ✕

AccessToken (http, Bearer)

PSD2 API OAuth2 access token can be generated by PSD2 OAuth2 API.

Value:

Authorize

7. Postup přístupu do aplikace přes konzoli na API portálu

Volání mocku PIS „New Standing Order initiation“ k otestování

Tato operace umožňuje založit nový trvalý příkaz. Po kliknutí na komponentu se rozbalí daná operace.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers
 Authorize

PISP ^

POST	/standingorders New standing order initiation	createNewStandingOrder
POST	/standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order	authorizeStandingOrder
GET	/standingorders/{transactionIdentification}/status Standing orders status	getStandingOrderStatus
GET	/standingorders/{transactionIdentification} Standing orders detail	getStandingOrderById
POST	/batchpayments New batch payment - instruction initiation	batchPaymentInitiation
POST	/batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization - starting the particular authorization method	batchAuthorizationWithCode
GET	/batchpayments/{transactionIdentification}/status Batch payment status	batchPaymentsStatus

Vyplnění požadovaných polí u operace „New Standing Order initiation“

Uživatel, který si přeje iniciovat nový trvalý příkaz, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněné pole je červeně zvýrazněno.

POST /standingorders New standing order initiation
createNewStandingOrder

Cancel

Name	Description
x-request-id string <small>(header)</small>	External Request ID <input style="width: 100%;" type="text" value="x-request-id"/>
TPP-Name * required string <small>(header)</small>	Transaction initiator name <input style="width: 100%;" type="text" value="TPP-Name"/>

application/json

Request Body

```

{
  "standingOrderIdentification": {
    "instructionIdentification": "NOTPROVIDED"
  },
  "amount": {
    "instructedAmount": {
      "value": 11.55,
      "currency": "CZK"
    }
  },
  "standingorder": {
    "execution": {
      "mode": "UNTIL_DATE",
      "modeDue": "DUE_DAY_OF_MONTH",
      "interval": "MONTHLY",
      "intervalDue": "30"
    },
    "exceptions": {
      "stoppages": "",
      "break": "r

```

Execute

Vypsání chyby u operace “New Standing Order initiation“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid No links</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>
401	<p>Missing certificate or access token No links</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>
403	<p>Invalid certificate or token No links</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>
415	<p>Invalid message charset No links</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>
500	<p>Unexpected error occurred No links</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>

Volání mocku PIS „Standing Orders Authorization“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Standing Orders Authorization“. Tato operace umožňuje nastartovat konkrétní autorizační metody. Po kliknutí na komponentu se rozbalí daná operace.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers
 Authorize

PISP

POST	/standingorders	New standing order initiation	createNewStandingOrder	🔒	⌵
POST	/standingorders/{transactionIdentification}/sign/{signId}	Request for authorization of standing order	authorizeStandingOrder	🔒	⌵
GET	/standingorders/{transactionIdentification}/status	Standing orders status	getStandingOrderStatus		⌵
GET	/standingorders/{transactionIdentification}	Standing orders detail	getStandingOrderById	🔒	⌵
POST	/batchpayments	New batch payment - instruction initiation	batchPaymentInitiation	🔒	⌵
POST	/batchpayments/{transactionIdentification}/sign/{signId}	Batch Authorization - starting the particular authorization method	batchAuthorizationWithCode	🔒	⌵
GET	/batchpayments/{transactionIdentification}/status	Batch payment status	batchPaymentsStatus		⌵

Vyplnění požadovaných polí u operace „Standing Orders Authorization“

Uživatel, který si přeje nastartovat autorizační proces, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněné pole je červeně zvýrazněno.

POST /standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order
authorizeStandingOrder

Cancel

Name	Description
transactionIdentification * required string <small>(path)</small>	Unique bank transaction identification <ul style="list-style-type: none"> example: '293748asdnjfbas20492asdioa'
<input type="text" value="transactionIdentification"/>	
signId * required string <small>(path)</small>	Identifier of the change
<input type="text" value="signId"/>	
x-request-id string <small>(header)</small>	External Request Id
<input type="text" value="x-request-id"/>	
TPP-Name * required string <small>(header)</small>	Transaction Initiator name
<input type="text" value="TPP-Name"/>	

Request body * required application/json

Initiation request - identifier of the authorization method and back URL

```
{
  "authorizationType": "USERAGENT_REDIRECT",
  "redirectUrl": "string"
}
```

Execute

Vypsání chyby u operace „Standing Orders Authorization“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

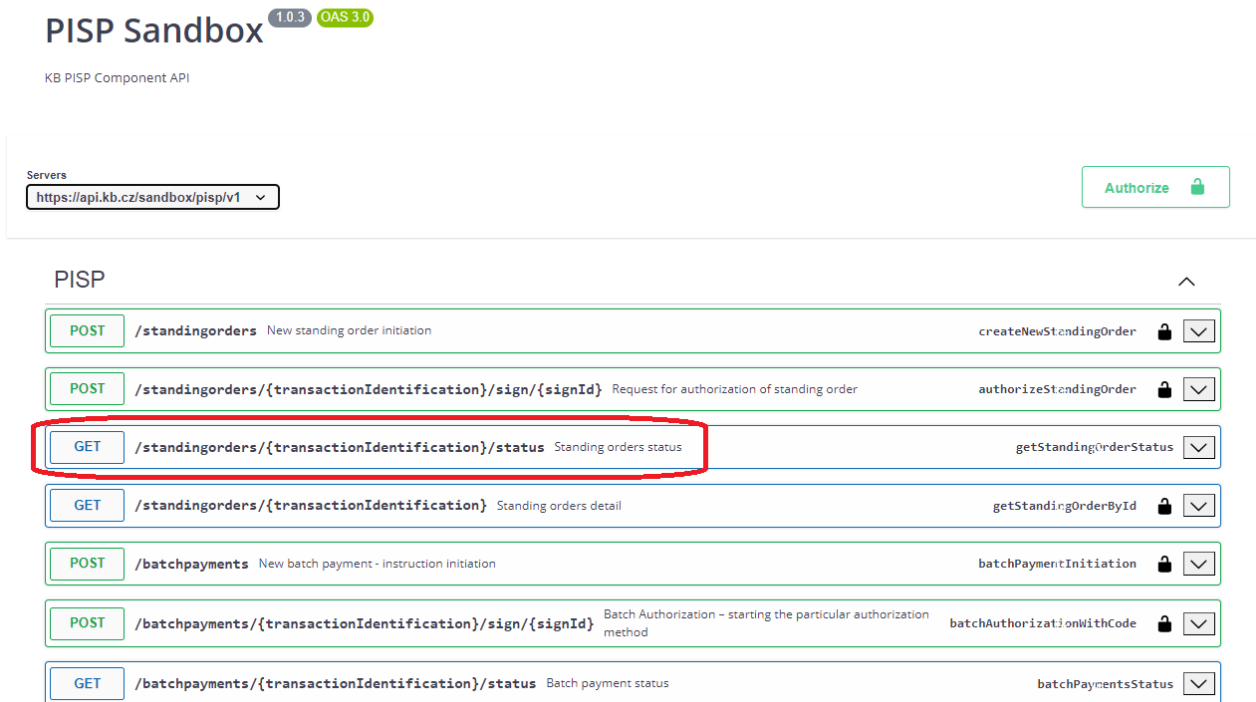
400	<p>Input parameter is invalid</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
404	<p>Id does not exist</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>	No links
415	<p>Invalid message charset</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR18", "message": "InvalidCharacterSet" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

Volání mocku PIS „Standing Order Status“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Standing Order Status“. Tato operace umožňuje zobrazit informace o stavu trvalém příkazu.

Endpoints

[Swagger download](#) | [Copy base URL](#)



PISP Sandbox 1.0.3 OAS 3.0
KB PISP Component API

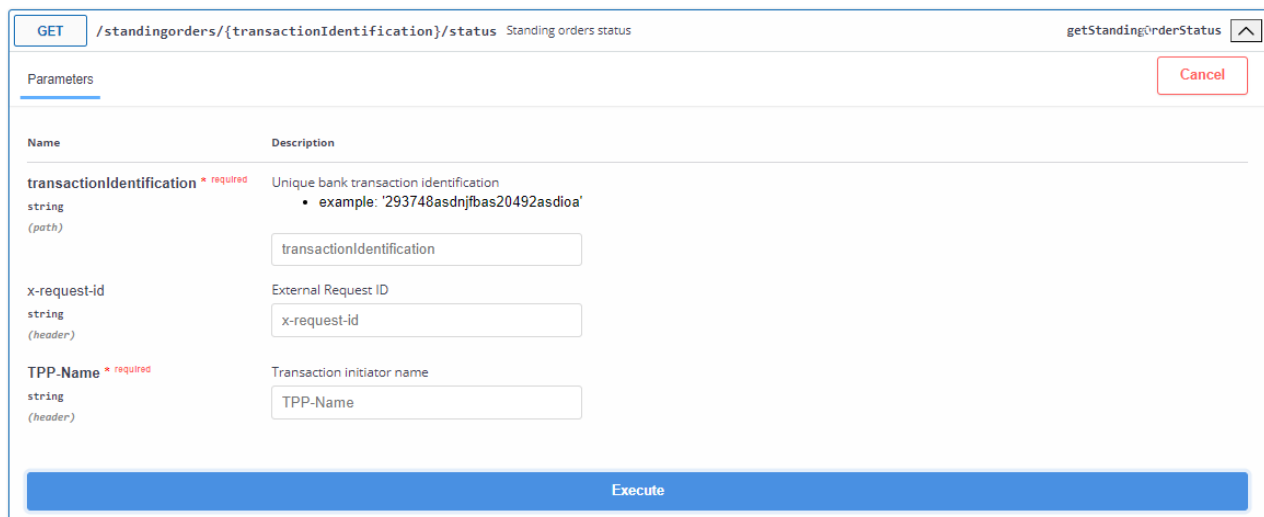
Servers: Authorize

PISP

- POST** /standingorders New standing order initiation createNewStandingOrder lock dropdown
- POST** /standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order authorizeStandingOrder lock dropdown
- GET** /standingorders/{transactionIdentification}/status Standing orders status getStandingOrderStatus dropdown
- GET** /standingorders/{transactionIdentification} Standing orders detail getStandingOrderById lock dropdown
- POST** /batchpayments New batch payment - instruction initiation batchPaymentInitiation lock dropdown
- POST** /batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization - starting the particular authorization method batchAuthorizationWithCode lock dropdown
- GET** /batchpayments/{transactionIdentification}/status Batch payment status batchPaymentsStatus dropdown

Vyplnění požadovaných polí u operace „Standing Order Status“

Uživatel, který si přeje zobrazit stav určité platby, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl, zobrazí se mu informace o stavu trvalého příkazu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou červeně zvýrazněna.



GET /standingorders/{transactionIdentification}/status Standing orders status getStandingOrderStatus dropdown

Parameters Cancel

Name	Description
transactionIdentification * required string (path)	Unique bank transaction identification • example: '293748asdnjfbas20492asdioa'
x-request-id string (header)	External Request ID
TPP-Name * required string (header)	Transaction initiator name

Execute

Vypsání chyby u operace „Standing Order Status“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
404	<p>Id does not exist</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>	No links
415	<p>Invalid message charset</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

Volání mocku PIS „Standing Order Detail“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Standing Order Detail“. Tato operace umožňuje zobrazit podrobnosti o trvalém příkazu.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers: Authorize

PISP ^

POST	/standingorders New standing order initiation	createNewStandingOrder
POST	/standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order	authorizeStandingOrder
GET	/standingorders/{transactionIdentification}/status Standing orders status	getStandingOrderStatus
GET	/standingorders/{transactionIdentification} Standing orders detail	getStandingOrderById
POST	/batchpayments New batch payment - instruction initiation	batchPaymentInitiation
POST	/batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization - starting the particular authorization method	batchAuthorizationWithCode
GET	/batchpayments/{transactionIdentification}/status Batch payment status	batchPaymentsStatus

Vyplnění požadovaných polí u operace „Standing Order Detail“

Uživatel, který si přeje zobrazit stav určité platby, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl, zobrazí se mu podrobnosti o trvalém příkazu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou červeně zvýrazněna.

GET /standingorders/{transactionIdentification} Standing orders detail getStandingOrderById

Cancel

Parameters

Name	Description
transactionIdentification * required string (path)	Unique bank transaction identification • example: '293748asdnjfbas20492asdioa'
<input type="text" value="transactionIdentification"/>	
TPP-Name * required string (header)	
<input type="text" value="TPP-Name"/>	
x-request-id string (header)	
<input type="text" value="x-request-id"/>	

Execute

Vypsání chyby u operace „Standing Order Status“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
415	<p>Invalid message charset</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

Volání mocku PIS „New batch payment - instruction initiation“ k otestování

Tato operace umožňuje založit dávkový příkaz. Po kliknutí na komponentu se rozbalí daná operace.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers: https://api.kb.cz/sandbox/pisp/v1 Authorize

PISP ^

- POST /standingorders New standing order initiation createNewStandingOrder
- POST /standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order authorizeStandingOrder
- GET /standingorders/{transactionIdentification}/status Standing orders status getStandingOrderStatus
- GET /standingorders/{transactionIdentification} Standing orders detail getStandingOrderById
- POST /batchpayments New batch payment - instruction initiation batchPaymentInitiation
- POST /batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization - starting the particular authorization method batchAuthorizationWithCode
- GET /batchpayments/{transactionIdentification}/status Batch payment status batchPaymentsStatus

Vyplnění požadovaných polí u operace „New batch payment - instruction initiation“

Uživatel, který si přeje založit dávkový příkaz, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněné pole je červeně zvýrazněno.

POST /batchpayments New batch payment - instruction initiation batchPaymentInitiation

[Cancel](#)

Parameters

Name	Description
x-request-id string <small>(header)</small>	External Request ID <input style="width: 100%;" type="text" value="x-request-id"/>

Request body required application/json

```

New batch payment request
{
  "exchangeIdentification": "103149078",
  "instructionName": "Rzdy březen 2021 z KB účtu.",
  "payments": [
    "ZUB0000K35E",
    "ZUB0000K35E"
  ]
}
    
```

Execute

Vypsání chyby u operace „New batch payment - instruction initiation“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
415	<p>Invalid message charset</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type <input type="text" value="application/json"/></p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

Volání mocku PIS „Batch Authorization – starting the particular authorization method“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Batch Authorization – starting the particular authorization method“ Tato operace umožňuje nastartovat konkrétní autorizační metody. Po rozkliknutí se rozbálí daná operace.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox

KB PISP Component API

1.0.3
OAS 3.0

Servers

https://api.kb.cz/sandbox/pisp/v1
▼

Authorize
🔒

PISP ^

POST	/standingorders <small>New standing order initiation</small>	createNewStandingOrder 🔒
POST	/standingorders/{transactionIdentification}/sign/{signId} <small>Request for authorization of standing order</small>	authorizeStandingOrder 🔒
GET	/standingorders/{transactionIdentification}/status <small>Standing orders status</small>	getStandingOrderStatus
GET	/standingorders/{transactionIdentification} <small>Standing orders detail</small>	getStandingOrderById 🔒
POST	/batchpayments <small>New batch payment - instruction initiation</small>	batchPaymentInitiation 🔒
POST	/batchpayments/{transactionIdentification}/sign/{signId} <small>Batch Authorization – starting the particular authorization method</small>	batchAuthorizationWithCode 🔒
GET	/batchpayments/{transactionIdentification}/status <small>Batch payment status</small>	batchPaymentsStatus

Vyplnění požadovaných polí u operace „Batch Authorization – starting the particular authorization method“

Uživatel, který si přeje nastartovat autorizační proces, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněné pole je červeně zvýrazněno.

POST
/batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization – starting the particular authorization method
batchAuthorizationWithCode

Cancel

Name	Description
x-request-id string <small>(header)</small>	External Request ID <input style="width: 100%; border: 1px solid #ccc;" type="text" value="x-request-id"/>
transactionIdentification * required string <small>(path)</small>	Batch transaction identification. Example: 9d7f87483ce8450cb85739691dfe574b <input style="width: 100%; border: 1px solid #ccc;" type="text" value="transactionIdentification"/>
signId * required string <small>(path)</small>	Identifier of the authorising process of the particular batch transaction. Example: 9d7f87483ce8450cb85739691dfe574b <input style="width: 100%; border: 1px solid #ccc;" type="text" value="signId"/>

Request body * required application/json ▼

Initiation request - identifier of the batch authorization method and back URL

```

{
  "authorizationType": "USERAGENT_REDIRECT",
  "redirectUrl1": "http://kb.cz",
  "batchDigest": "AAF21A859D225FD1D188982F08DA3E4A9AD6F7CC"
}
          
```

Execute

Vypsání chyby u operace „Batch Authorization – starting the particular authorization method“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
404	<p>Id does not exist</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>	No links
415	<p>Invalid message charset</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

Volání mocku PIS „Batch payment status“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Batch payment status“. Tato operace umožňuje zobrazit informace o stavu dávkového příkazu.

Endpoints

[Swagger download](#) | [Copy base URL](#)

PISP Sandbox 1.0.3 OAS 3.0

KB PISP Component API

Servers: Authorize

PISP ^

- POST /standingorders New standing order initiation createNewStandingOrder
- POST /standingorders/{transactionIdentification}/sign/{signId} Request for authorization of standing order authorizeStandingOrder
- GET /standingorders/{transactionIdentification}/status Standing orders status getStandingOrderStatus
- GET /standingorders/{transactionIdentification} Standing orders detail getStandingOrderById
- POST /batchpayments New batch payment - instruction initiation batchPaymentInitiation
- POST /batchpayments/{transactionIdentification}/sign/{signId} Batch Authorization - starting the particular authorization method batchAuthorizationWithCode
- GET /batchpayments/{transactionIdentification}/status Batch payment status batchPaymentsStatus

Vyplnění požadovaných polí u operace „Batch payment status“

Uživatel, který si přeje zobrazit stav určitého dávkového příkazu, klikne na tlačítko „Try it out“ a vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl, zobrazí se mu podrobnosti o dávkové platbě. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou červeně zvýrazněna.

GET /batchpayments/{transactionIdentification}/status Batch payment status batchPaymentsStatus

Resource for getting status of batch and the batch payment

Parameters Cancel

Name	Description
transactionIdentification * required string (path)	Batch transaction identification. Example: 9d7f87483ce8450cb85739691dfe574b <input style="width: 100%;" type="text" value="transactionIdentification"/>
x-request-id string (header)	External Request ID <input style="width: 100%;" type="text" value="x-request-id"/>

Execute

Vypsání chyby u operace „Batch payment status“

Pokud byla některá hodnota zadaná špatně, vypíše se po zmáčknutí tlačítka „Execute“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

400	<p>Input parameter is invalid</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	No links
401	<p>Missing certificate or access token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	No links
403	<p>Invalid certificate or token</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	No links
500	<p>Unexpected error occurred</p> <p>Media type application/json</p> <p>Example Value Schema</p> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	No links

8. Postup přístupu do aplikace přes přímé volání

Nový trvalý příkaz – iniciace platby (POST/my/standingorders)

Resource pro založení nového trvalého příkazu.

Charakteristika resource

URI: /standingorders
HTTP Metoda: POST
Request URL: <https://api.kb.cz/sandbox/pisp/v1/standingorders/>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“ V tomto poli jsou podporovány pouze znaky bez diakritiky.
TPP-Identification	Text	Ne	Identifikace původního TPP, které request vytvořilo. Např.: „CZ013574-15“

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json.

Obsah POST requestu a response volání viz kapitola 0 Elementy zprávy Nová platba – iniciace trvalého příkazu

Chybové kódy definované pro službu POST iniciace trvalého příkazu

HTTP CODE	STATUS	ERROR CODE	ÚČEL
401		UNAUTHORISED	Invalid/missing access token = user is not authenticated
401		UNAUTHORISED	Invalid/missing certificate = provider is not authenticated
403		FORBIDDEN	Calling of the method which does not correspond to the licence, or invalid certificate.
400		FIELD_MISSING	Missing mandatory field in the request.
400		FIELD_INVALID	FIELD value is not valid.
400		AC02	[InvalidDebtorAccountNumber] – invalid account identifier in the request content.

400	AC03	[InvalidCreditorAccountNumber] - recipient account number is closed, blocked, credit card are disabled for the account type, or the recipient's account number is in invalid format (note: only valid for in-house payments). Není používáno.
400	AC09	[InvalidAccountCurrency] – the specified payer account currency does not correspond to the client's account currency for the account number held in the bank (the account currency is optional, in the case of multicurrency accounts, the account currency must be specified - Raiffeisenbank). Využíváno namísto AC10.
400	AC12	[InvalidAccountType] - account type does not match allowed account types (e.g., a non-paying account).
400	AM11	[InvalidTransactionCurrency] – the request contains a currency not trade/not supported.
400	AM12	[InvalidAmount] – wrong amount. For instance, too low or high amount or wrong number format according to the number of decimal places according to the ISO 4217.
400	DT01	[InvalidDate] - non-existent date value or format. Maturity in the future or in the past. Transactions on a non-business day of a bank (unless the bank receives such orders).
400, 50x	NARR	Narrative – a general reason for rejecting the payment, with an addition of error-related information.
400	RR10	[InvalidCharacterSet] – invalid character set in the request.
400	REC_SEND	[RECEIVER_SAME_AS_SENDER] – Payer's account is the same as the payee's account (Applies to all types of payments)

Elementy zprávy Nová platba – iniciace trvalého příkazu

Uvažované typy trvalých příkazů

Trvalé příkazy je povoleno zadávat výhradně jako tuzemskou platbu na území ČR.

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZANTACE
+	standingOrderIdentification	[1..1]	ALL	Max35Text	Standing order identification
++	instructionIdentification	[1..1]	ALL	Max35Text	Instruction identification
++	endToEndIdentification	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	Max35Text	End to end identification. SEPA – Není podporováno
++	transactionIdentification	[0..0]	ALL	Max35Text	Transaction identification
+	paymentTypeInfoInformation	[0..1]	ALL	The same object as to initiate payments chapter ...	
+	amount	[1..1]	ALL	STDO - AmountType3CZ ESST - AmountType3CZ XBST AmountType3Choice	Payment amount
++	instructedAmount	[1..1]	ALL	CurrencyAndAmount	Amount and currency in the instruction
+++	value	[0..0]	ALL	Amount	Amount of the transfer. Není podporováno.
+++	currency	[0..0]	ALL	CurrencyCode	Transfer Currency. Není podporováno.
+	requestedExecutionDate	[1..1]	ALL	ISODate	Requested date of payment
+	standingOrder	[1..1]	ALL	Structure	Structure describing the parameters of the standing orders.

++	alias	[0..1]	ALL	Max250Text	Description resp. the user-defined standing order payment name. Toto pole není na vstupu podporováno. Pro alias je v KB využívána hodnota Remittanceinformation ze které je informace přejata do pole alias
++	execution	[1..1]	ALL		Features of execution a standing order for payment.
+++	mode	[1..1]	ALL	Max35Text	The execution mode defines when or how standing order will be cancelled, processed the last time. Possible values: UNTIL_DATE (standing order is valid until specific date - field lastExecutionDate), UNTIL_CANCELLATION (standing order is valid forever and must be cancelled by client), AFTER_MAX_ITERATION_EXCEEDED (certain count of executions is specified - field maxIterations) V KB je u definovaného počtu opakování vypočteno a doplněno datum poslední platby or MAX_AMOUNT_EXCEEDED (maximum amount which can be transferred for this order is specified, if next iteration would exceed this amount it is not executed - field maxAmount). Není podporováno
+++	modeDue	[1..1]	ALL	Max35Text	The execution due mode defines how the date when order should be executed is specified. Possible values: DUE_DAY_OF_MONTH (specific number of day in the month is defined) or DUE_OR_BEFORE_DAY_OF_MONTH (is defined as a specific day of the month or the previous day if the day falls on a non-banking day) or DUE_OR_NEXT_DAY_OF_MONTH (is defined as a specific day of the month or the next day if the day falls on a non-banking day) or DUE_LAST_DAY_OF_MONTH (order is executed

					on last day of particular month). Není podporováno
+++	interval	[1..1]	ALL	Max10Text	Execution interval defines how often order is executed. Possible values: DAILY, WEEKLY, BI_WEEKLY, MONTHLY, BI_MONTHLY, QUARTERLY, HALFYEARLY, YEARLY, IRREGULAR — Není podporováno.
+++	intervalDue	[0..0]	ALL	Number	Value represents order number of the day within particular period when the standing order will be regularly executed. Possible values: 1-7 (for WEEKLY interval), 1-28 for MONTHLY, 1-2 for BI_MONTHLY, 1-3 for QUARTERLY, 1-6 for HALFYEARLY, 1-12 for YEARLY. Není podporováno
++	validity	[0..1]	ALL		Elements restricting the validity of the standing order.
+++	lastExecutionDate	[0..1]	ALL	ISODate	Date when the last order will be processed. Null value responds to ending by the user.
+++	maxIterations	[0..1]	ALL	Number	Maximum number of iterations - processing of the standing order. Only applicable in combination with executionMode AFTER_MAX_ITERATION_EXCEEDED.
+++	maxAmount	[0..0]	ALL	Amount	Maximum amount to be transferred using the standing order. Only applicable in combination with executionMode AFTER_MAX_AMOUNT_EXCEEDED. Není podporováno
++++	value	[0..0]	ALL	Amount	Maximum amount of the transfer. Není podporováno
++++	currency	[0..0]	ALL	CurrencyCode	Transfer Currency of the maximum amount. Není podporováno
++	exceptions	[0..0]	ALL		Elements defining exceptions to executing a standing order. Není podporováno.
+++	stoppages	[0..0]	ALL	Array of Max20Text	List of months where there is no payment. Possible values: JANUARY, FEBRUARY,

					MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER. Není podporováno
+++	breaks	[0..0]	ALL	Array of Intervals	Break periods. Není podporováno.
++++	validFromDate	[0..0]	ALL	ISODate	Start date of one break period. Standing order will not be processed from this date. Není podporováno.
++++	validToDate	[0..0]	ALL	ISODate	End date of one break period. Standing order will not be processed to this date. Není podporováno.
+	exchangeRateInformation	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	chargeBearer	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	ChargeBearerType1Code	Fee payer. Na vstupu je provedena validace oproti zadaným hodnotám KB s hodnotami dále nepracuje. EHP; NONEHP – Není podporováno
+	chargesAccount	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	ultimateDebtor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA – Není podporováno
+	debtor	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	debtorAccount	[1..1]	ALL	The same object as to initiate payments chapter ...	
+	intermediaryAgent1	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	creditorAgent	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA; EHP; NONEHP – Není podporováno
+	creditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA; EHP; NONEHP – Není podporováno
+	creditorAccount	[1..1]	ALL	The same object as to initiate payments chapter ...	
+	ultimateCreditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA – Není podporováno
+	Purpose	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	instructionForNextAgent	[0..0]	ALL	Instruction code	Instructions for the next bank

+	remittanceInformation	[0..1]	ALL	The same object as to initiate payments chapter ...
---	-----------------------	--------	-----	---

V případě, že v poli **remittanceInformation.structured.creditorReferenceInformation.reference** bude obsahovat variabilní, konstantní nebo specifický symbol, dojde k jejich rozpoznání a samostatnému uložení do příslušných polí pro tyto symboly v modelu PISP.

- Hodnota **variabilního symbolu** se zaznamenává jako VS:max.10 číslic (např. VS:3451859072).
- Hodnota **konstantního symbolu** se zaznamenává jako KS:max.10 znaků (např. KS:0308).
- Hodnota **specifického symbolu** se zaznamenává jako SS:max.10 číslic (např. SS:8451201274).

JSON příklad elementu:

```
"reference": "VS:123456\", \"KS:456789\", \"SS:879213546"
```

pozn. k poli remittanceInformation.unstructured: dle standardu bankovní asociace může i toto pole obsahovat informace o VS, KS a SS, nicméně my budeme s veškerými informacemi zde obsaženými pracovat tak, jako by se jednalo o prostý popis platby (informace pro příjemce) a k žádnému parsování symbolů nebude docházet, ani kdyby se zde vyskytly.

Elementy response Nový trvalý příkaz – iniciace trvalého příkazu

Tabulka obsahuje jen elementy, které se vyskytují pouze v response zprávě.

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	standingOrderIdentification	[1..1]	ALL	Max35Text
++	transactionIdentification	[1..1]	Max35Text	Identifier of established standing order instruction.
+	paymentTypeInfoInformation	[1..1]	ALL	
++	serviceLevel	[1..1]	±	Service placement (within type of payment).
+++	code	[1..1]	Text	Type of entered payment. Je podporováno pouze DMCT - DomesticPayment
+	signInfo	[1..1]	±	Status information and id of unauthorized transactions
++	state	[1..1]	StateCode	Transaction authorization status V KB je vždy vrácen stav OPEN
++	signId	[1..1]	Text	Identifier of the authorization process of a particular transaction.
+	instructionStatus	[1..1]	StatusCode	Status of entered standing order V KB jsou vráceny stavy ACWC nebo ACTC
+	statusChangeInfo	[0..1]	Text	Specifies the change which was made to the instruction. Used with instructionStatus "ACWC".

Hodnoty elementu serviceLevel.code – Typ iniciované platby

KÓD	POPIS
DMCT	[DoMestic Credit Transfer] Domestic payment
ESCT	[SEPA Credit Transfer] – SEPA payment. Není podporováno.
XBCT	[Cross-Border Credit Transfer] – Cross border payment. Není podporováno
EXCT	[EEA X-border Credit Transfers] – Cross border payment within the EEP. Není podporováno
NXCT	[Non-EEA X-border Credit Transfers] – Cross border payment outside the EEP. Není podporováno.

Status kódy plateb – StatusCode

HTTP STATUS KÓD	STATUS CODE	ÚČEL
200	ACTC	[AcceptedTechnicalValidation] - Authentication and syntactical and semantical validation are successful
200	RJCT	[Rejected] - Payment initiation or individual transaction included in the payment initiation has been rejected
200	ACWC	[AcceptedWithChange] - Instruction is accepted but a change will be made, such as date or remittance not sent

Iniciace autorizace trvalé platby – specifické pro každou banku (POST / standingorders/{transactionIdentification}/sign/{signId})

Tento resource je určen ke **startu konkrétní autorizační metody** z vybraného scénáře.

Vstupem je JSON objekt obsahující požadovaný typ autorizační metody - **CODE** a všechny elementy specifické pro tento krok.

Výstupem tohoto resource je přehled hodnot potřebných pro dokončení autorizace.

Např. pro CODE odpovídající federované autorizaci bude odpovědí URL a parametry pro přesměrování na federovanou autorizační stránku.

A např. pro CODE odpovídající autorizaci prostřednictvím OTP kódu zaslání SMS bude odpovědí jen potvrzení o odeslání kódu. Samotné odeslání iniciuje banka.

Požadavek na autorizaci trvalého příkazu je nutné poslat nejpozději do 5-ti minut od zaslání iniciace platby třetí stranou.

Charakteristika resource

URI: standingorders/{transactionIdentification}/sign/{signId}
HTTP Metoda: POST
Request URL: <https://api.kb.cz/sandbox/pisp/v1/standingorders/{transactionIdentification}/sign/{signId}>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.

Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“. V tomto poli jsou podporovány pouze znaky bez diakritiky.
TPP-Identification	Text	Ne	Identifikace původního TPP, které request vytvořilo. Např.: „CZ013574-15“

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Obsah POST requestu a response volání viz kapitola 0 ELEMENTY ZPRÁVY Krok II. Inicie autorizace trvalého příkazu – specifické pro každou banku

Chybové kódy definované pro službu POST Autorizace trvalého příkazu:

HTTP STATUS KÓD	ERROR CODE	ÚČEL
401	UNAUTHORISED	Invalid/missing access token = user is not authenticated
403	FORBIDDEN	Invalid/missing certificate = provider is not authenticated
404	ID_NOT_FOUND	The required id does not exist

Elementy zprávy Inicie autorizace trvalého příkazu

Parametry requestu:

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁT	ÚČEL
+	authorizationType	[1..1]	Text	Code of required authorization (from authorization scenarios)
+	redirectUrl	[0..1]	Text	URL link or package of federated authorization call back address. This address is used by the federated bank authorization to redirect back to the TPP application after authorization is complete.

Parametry response iniciace autorizace trvalého příkazu:

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PREZENTACE FORMÁTU	PREZENTACE
+	authorizationType	[1..1]	±	Code of required authorization (from authorization scenarios)
+	Href	[1..1]	±	Reference to call federated authorization
++	url	[1..1]	Text	URL link or package of federated authorization. Vždy vráceno.
++	id	[1..1]	Text	Possible id for calling federated authorization. Vždy vráceno
+	method	[1..1]	Text	Method to use href link and federated authorization. Vždy vráceno.
+	formData	[0..1]	±	Optional element. For the method POST federated authorization (authorizationType=USERAGENT_REDIRECT) element Contains data for sending in redirecting to federated authorization.

++	SAMLRequest	[0..1]	Text	Optional parameter. For the method POST federated authorization. (authorizationType=USERAGENT_REDIRECT) element Contains data of SAML request. KB využívá pouze GET.
++	relayState	[0..1]	Text	Optional parameter. For the method POST federated authorization (authorizationType=USERAGENT_REDIRECT) element Contains relayState for returnable value. KB využívá pouze GET.
+	signInfo	[1..1]	±	Information on instruction author
++	state	[1..1]	Text	Status of transaction authorization in a format supported by the bank
++	signId	[1..1]	Text	Unique identifier for current transaction authorization

Detail trvalého příkazu (GET /my/standingorders/{transactionIdentification})

Resource pro zobrazení informace o založeném trvalém příkazu. Jedná se o trvalý příkaz, který byl přijat k autorizaci, ale ještě nebyl klientem autorizován. Resource pracuje pouze s transakcemi založenými prostřednictvím konkrétního providera.

Resource pro zjištění detailu transakce. Informace o providerovi je přebírána z certifikátu, resp. informaci o licenci.

Charakteristika resource

URI: /standingorders/{transactionIdentification}
HTTP Metoda: GET
Request URL: <https://api.kb.cz/sandbox/pisp/v1/standingorders/{transactionIdentification}>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu:

PARAMET	TYP	POVINNÝ	ÚČEL
transactionIdentification	Max35Text	Yes	Identifier of established standing order instruction

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“. V tomto poli jsou podporovány pouze znaky bez diakritiky.
TPP-Identification	Text	Ne	Identifikace původního TPP, které request vytvořilo. Např.: „CZ013574-15“

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Elementy zprávy Detail trvalého příkazu

PARAMETER	TYPE	MANDATORY	PURPOSE
transactionIdentification	Max35Text	Yes	Identifier of established standing order instruction

Chybové kódy definované pro službu GET Detail trvalého příkazu

HTTP STATUS CODE	CHYBOVÝ KÓD	ÚČEL
401	UNAUTHORISED	Invalid/missing access token = user is not authenticated
401	UNAUTHORISED	Invalid/missing certificate = provider is not authenticated
404	ID_NOT_FOUND	Neplatné nebo neznámé ID účtu

Elementy response zprávy Detail trvalého příkazu

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FOTMÁRU	PREZENTACE
+	standingOrderIdentification	[1..1]	ALL	Max35Text	Standing order identification
++	instructionIdentification	[1..1]	ALL	Max35Text	Instruction identification
++	endToEndIdentification	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	Max35Text	End to end identification SEPA – Není podporováno
++	transactionIdentification	[0..0]	ALL	Max35Text	Transaction identification
+	paymentTypeInformation	[1..1]	ALL	The same object as to initiate payments chapter ...	
+	amount	[1..1]	ALL	STDO - AmountType3CZ ESST - AmountType3CZ XBST AmountType3Choice	Payment amount
++	instructedAmount	[1..1]	ALL	CurrencyAndAmount	Amount and currency in the instruction
+++	value	[1..1]	ALL	Amount	Amount of the transfer
+++	currency	[1..1]	ALL	CurrencyCode	Transfer Currency
+	requestedExecutionDate	[1..1]	ALL	ISODate	Requested date of payment
+	standingOrder	[1..1]	ALL	Structure	Structure describing the parameters of the standing orders.
++	alias	[0..1]	ALL	Max250Text	Description resp. the user-defined standing order payment name. V poli alias je vždy zobrazena informace z pole remittanceinformation.

++	execution	[1..1]	ALL		Features of execution a standing order for payment.
+++	mode	[1..1]	ALL	Max35Text	The execution mode defines when or how standing order will be cancelled, processed the last time. Possible values: UNTIL_DATE (standing order is valid until specific date - field lastExecutionDate), UNTIL_CANCELLATION (standing order is valid forever and must be cancelled by client), AFTER_MAX_ITERATION_EXCEEDED (certain count of executions is specified - field maxIterations) podle počtu opakování je bankou automaticky doplněno datum poslední platby or MAX_AMOUNT_EXCEEDED (maximum amount which can be transferred for this order is specified, if next iteration would exceed this amount it is not executed - field maxAmount). Není podporováno
+++	modeDue	[1..1]	ALL	Max35Text	The execution due mode defines how the date when order should be executed is specified. Possible values: DUE_DAY_OF_MONTH (specific number of day in the month is defined) or DUE_OR_BEFORE_DAY_OF_MONTH (is defined as a specific day of the month or the previous day if the day falls on a non-banking day) or DUE_OR_NEXT_DAY_OF_MONTH (is defined as a specific day of the month or the next day if the day falls on a non-banking day) or DUE_LAST_DAY_OF_MONTH (order is executed on last day of particular month). Není podporováno
+++	interval	[1..1]	ALL	Max10Text	Execution interval defines how often order is executed. Possible values: DAILY, WEEKLY, BI-WEEKLY, MONTHLY, BI_MONTHLY, QUARTERLY,

					HALFYEARLY, YEARLY, IRREGULAR — Není podporováno.
+++	intervalDue	[0..0]	ALL	Number	Value represents order number of the day within particular period when the standing order will be regularly executed. Possible values: 1-7 (for WEEKLY interval), 1-28 for MONTHLY, 1-2 for BI_MONTHLY, 1-3 for QUARTERLY, 1-6 for HALFYEARLY, 1-12 for YEARLY. Není podporováno.
++	validity	[1..1]	ALL		Elements restricting the validity of the standing order. V případě, že není vyplněn datum konečné splatnosti, je vráceno vždy prázdné pole.
+++	lastExecutionDate	[0..1]	ALL	ISODate	Date when the last order will be processed. Null value responds to ending by the user.
+++	maxIterations	[0..0]	ALL	Number	Maximum number of iterations - processing of the standing order. Only applicable in combination with executionMode AFTER_MAX_ITERATION_EXCEEDED. Není podporováno
+++	maxAmount	[0..0]	ALL	Amount	Maximum amount to be transferred using the standing order. Only applicable in combination with executionMode AFTER_MAX_AMOUNT_EXCEEDED. Není podporováno.
++++	value	[1..1]	ALL	Amount	Maximum amount of the transfer
++++	currency	[1..1]	ALL	CurrencyCode	Transfer Currency of the maximum amount
++	exceptions	[0..1]	ALL		Elements defining exceptions to executing a standing order.
+++	stoppages	[0..0]	ALL	Array of Max20Text	List of months where there is no payment (only applicable with interval IRREGULAR). Possible values: JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER. Není podporováno

+++	breaks	[0..1]	ALL	Array of Intervals	Break periods.
++++	validFromDate	[0..1]	ALL	ISODate	Start date of one break period. Standing order will not be processed from this date.
++++	validToDate	[0..1]	ALL	ISODate	End date of one break period. Standing order will not be processed to this date.
+	exchangeRateInformation	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	chargeBearer	[0..0] [0..0] [0..0] [0..1]	TUZEM SEPA EHP NONEHP	ChargeBearerType1Code	Fee payer. EHP - Pouze kontrola formátu zpracování není podporováno
+	chargesAccount	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	ultimateDebtor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA - Není podporováno
+	debtor	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	debtorAccount	[1..1]	ALL	The same object as to initiate payments chapter ...	
+	intermediaryAgent1	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	creditorAgent	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA, EHP, NONEHP - Není podporováno
+	creditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA, EHP, NONEHP - Není podporováno
+	creditorAccount	[1..1]	ALL	The same object as to initiate payments chapter ...	
+	ultimateCreditor	[0..0] [0..0] [0..0] [0..0]	TUZEM SEPA EHP NONEHP	The same object as to initiate payments chapter ...	SEPA - Není podporováno
+	purpose	[0..0]	ALL	The same object as to initiate payments chapter ...	
+	instructionForNextAgent	[0..0]	ALL	Instruction code	Instructions for the next bank
+	remittanceInformation	[1..1]	ALL	The same object as to initiate payments chapter ...	Je vraceno vždy i v případě, že nebylo nic zadáno.
+	signInfo	[1..1]	ALL	±	Status information and id of unauthorized transactions
++	state	[1..1]	ALL	StateCode	Transaction authorization status
++	signId	[1..1]	ALL	Text	Identifier of the authorization process of a particular transaction.

					Vráceno vždy.
+	instructionStatus	[1..1]	ALL	Status Code set	Transaction status identifier
+	statusChangeInfo	[0..0]	ALL	Text	Specifies the change which was made to the instruction. Used with instructionStatus "ACWC". Není podporováno

Status založeného/iniciovaného trvalého příkazu (GET /my/standingorders/{transactionIdentification}/status)

Resource pro získání stavu trvalého příkazu

Charakteristika resource

URI: /standingorders/{transactionIdentification}/stauts
HTTP Metoda: GET
Request URL: <https://api.kb.cz/sandbox/pisp/v1/standingorders/{transactionIdentification}/status>
Authorization: request **nevyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TPY	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“. V tomto poli jsou podporovány pouze znaky bez diakritiky.
TPP-Identification	Text	Ne	Identifikace původního TPP, které request vytvořilo. Např.: „CZ013574-15“

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Obsah POST requestu a response volání viz kapitola 0 Elementy zprávy Status založeného/iniciovaného trvalého příkazu

Chybové kódy definované pro službu GET Status založeného/iniciované trvalého příkazu

HTTP STATUS KÓD	CHYBOVÝ KÓD	ÚČEL
401	UNAUTHORISED	Nevalidní/chybějící certifikát = provider není autentizován
404	TRANSACTION_MISSING	Volání metody, která neodpovídá licenci, nebo neplatný certifikát.
404	ID_NOT_FOUND	Neplatné nebo neznámé ID účtu

Elementy zprávy Status založeného/iniciovaného trvalého příkazu

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORÁTU	PREZENTACE
+	instructionStatus	[1..1]	StatusCode	Status of entered standing order
+	realizedPayments	[0..0] conditional if it is an authorized standing order	Array	Array of transaction identifiers. Není podporováno
++	transactionIdentification	[1..1]	Max35Text	Identifier of established transaction
++	date	[0..1]	ISODate/ISODateTime	Due date/payment foreign currency in the format ISODate, , or ISODateTime, i.e. YYYY-MM-DD, or YYYY-MM-DDThh:mm:ss.STZD, depending on the transaction type and method how the bank presents data (and time) of due date/payment foreign currency. Mainly for card or cash transactions it is posted as ISODateTime.
+	errorinfo	[0..1]	Array	
++	error	[0..1]	String	
++	parameters	[0..1]	Array	
++	message	[0..1]	String	

Status kódy plateb – StatusCode

HTTP STATUS KÓD	STATUSCODE	ÚČEL
200	ACTC	[AcceptedTechnicalValidation] - Authentication and syntactical and semantical validation are successful
200	RJCT	[Rejected] - Payment initiation or individual transaction included in the
200	ACSP	[AcceptedSettlementInProgress] - All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution
200	ACSC	[AcceptedSettlementCompleted] – Standing order on the debtor's account has been completed

Dávková platba – iniciace (POST /my/batchpayments)

Resource pro založení dávkové platby.

Dodatečné informace:

- Dávková platba může samostatně zahrnovat transakce stejného typu tuzemské i SEPA platby (například: pouze SEPA, pouze Domáci).
- Okamžité a zahraniční platby nejsou podporovány.

Charakteristika resource

URI: /batchpayments
HTTP Metoda: POST
Request URL: <https://api.kb.cz/sandbox/pisp/v1/batchpayments>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“ V tomto poli jsou podporovány pouze znaky bez diakritiky.
X-Request-ID	Text	Ano	Unikátní identifikátor požadavku aplikace TPP.

Elementy request Dávková platba – iniciace

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	exchangeIdentification	[1..1]	Max14Text	Jednoznačná identifikace dotazu.
+	instructionName	[0..1]	Max35Text	Volitelné pojmenování dávky.
+	payments	[1..1]	Max35Text	Jednoznačná identifikace konkrétní transakce pro autorizační proces.

Elementy response Dávková platba – iniciace

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	transactionIdentification	[1..1]	Max35Text	Identifikátor založené dávky.
+	signInfo	[1..1]	±	Informace o stavu a id neautorizované dávky.
++	state	[1..1]	StateCode	Informace o stavu autorizace transakce.
++	signId	[1..1]	Text	Identifikátor autorizačního procesu konkrétní dávky.
+	instructionStatus	[1..1]	StatusCode	Status založené dávky
+	statusChangeInfo	[0..1]	Text	Určuje změnu, která byla provedena v instrukci. Používá se s instructionStatus "ACWC".
+	batchDigest	[1..1]	Max50Text	Unikátní ID, zabalené do batch SHA HASH, pomoci, kterého jde zkontrolovat, že mezi iniciací a autorizací dávky nedošlo k žádné změně.

Status codes of payment – StatusCode

HTTP STATUS CODE	STATUS CODE	PURPOSE
200	ACTC	[AcceptedTechnicalValidation] - Authentication and syntactical and semantical validation are successful
200	RJCT	[Rejected] - Payment initiation or individual transaction included in the payment initiation has been rejected
200	ACWC	[AcceptedWithChange] - Instruction is accepted but a change will be made, such as date or remittance not sent

Chybové kódy definované pro službu POST iniciace dávkové platby

HTTP STATUS CODE	ERROR CODE	PURPOSE
401	UNAUTHORISED	Chybí kvalifikovaný certifikát.
403	FORBIDDEN	Volání metody, která neodpovídá licenci, nebo neplatný kvalifikovaný certifikát.
404	[ID_NOT_FOUND]	V případě, že je pole vyplněno, ale je nevalidní.
404	NOT_FOUND	Pokud parametr není vyplněn nebo pole neodpovídá max. délce.
400	TYPE_DIFFERENT	Platby mohou být jednoho typu Domestic nebo jen SEPA.
400	FIELD_MISSING	Povinné pole nebylo vyplněno.
400	FIELD_INVALID	Hodnota pole není validní.
400	AC12	[InvalidAccountType] Typ účtu neodpovídá povoleným typům účtů. Např. se jedná o neplatební účet.
400	NARR	[Exceeded number of orders in a single batch]
400	NARR	[Batch payment is empty]
400	NARR	[ID is not unique]
400	NARR	[Foreign payment - unsupported payment type.]
400	NARR	[Transaction has already been authorized or rejected.]
400	NARR	[Daily limit of transactions in batches exhausted.]

Iniciace autorizace dávkové platby (POST/my/batchpayments/{transactionIdentification}/sign/{signId})

Tento resource je určen ke startu konkrétní autorizační metody z vybraného scénáře.

Vstupem je JSON objekt obsahující požadovaný typ autorizační metody - CODE a všechny elementy specifické pro tento krok.

Výstupem tohoto resource je přehled hodnot potřebných pro dokončení autorizace.

Např. pro CODE odpovídající federované autorizaci bude odpovědí URL a parametry pro přesměrování na federovanou autorizační stránku.

A např. pro CODE odpovídající autorizaci prostřednictvím OTP kódu zasláného SMS bude odpovědí jen potvrzení o odeslání kódu. Samotné odeslání iniciuje banka.

Charakteristika resource

URI: </batchpayments/{transactionIdentification}/sign/{signId}>
HTTP Metoda: POST
RequestURL: <https://api.kb.cz/sandbox/pisp/v1/batchpayments/{transactionIdentification}/sign/{signId}>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“ V tomto poli jsou podporovány pouze znaky bez diakritiky.
X-Request-ID	Text	Ano	Unikátní identifikátor požadavku aplikace TPP.

Path parametry:

PARAMETR	TYP	POVINNÝ	ÚČEL
transactionIdentification	Text	Ano	Identifikátor založené dávky.
signId	Text	Ano	Identifikátor autorizačního procesu konkrétní dávky.

Elementy request iniciace autorizace dávkové platby

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	authorizationType	[1..1]	Text	Kód požadované autorizace (z autorizačních scénářů) Povolena hodnota "USERAGENT_REDIRECT",
+	redirectUrl	[0..1]	Text	URL odkaz nebo balíček adresy pro zpětné volání federované autorizace. Tuto adresu používá autorizace federované banky k

				přesměrování zpět do aplikace TPP po dokončení autorizace.
+	batchDigest	[0..1]	Max50Text	Unikátní ID, zabalené do batch SHA HASH, pomocí kterého jde zkontrolovat, že mezi iniciací a autorizací dávky nedošlo k žádné změně. Doporučujeme jeho využití z pohledu bezpečnosti.

Elementy response iniciace autorizace dávkové platby

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZENTACE
+	authorizationType	[1..1]	PISP ALL	±	Kód požadované <i>autorizace</i> (z <i>autorizačních scénářů</i>)
+	href	[0..1]	PISP ALL	±	Reference pro vyvolání federované autorizace
++	url	[1..1]	PISP ALL	Text	URL link nebo package federované autorizace
++	id	[0..1]	PISP ALL	Text	Případné id pro vyvolání federované autorizace
+	method	[0..1]	PISP ALL	Text	Metoda pro použití href linku u federované autorizace.
+	formData	[0..1]	PISP ALL	±	Volitelný element. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje data pro odeslání v přesměrování na federovanou autorizaci.
++	SAMLRequest	[0..1]	PISP ALL	Text	Volitelný parametr. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje data SAML requestu.
++	relayState	[0..1]	PISP ALL	Text	Volitelný parametr. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje relayState pro návratovou hodnotu.
+	signInfo	[1..1]	PISP ALL	±	Informace o autorizaci instrukce.
++	state	[1..1]	PISP ALL	Text	Status autorizace transakce v bankou podporovaném formátu.
++	signId	[1..1]	PISP ALL	Text	Jedinečný identifikátor aktuální autorizace dávky.

Chybové kódy definované pro službu POST iniciace autorizace dávkové platby

HTTP STATUS CODE	ERROR CODE	PURPOSE
401	UNAUTHORISED	Chybí kvalifikovaný certifikát.
403	FORBIDDEN	Volání metody, která neodpovídá licenci, nebo neplatný kvalifikovaný certifikát.
404	[ID_NOT_FOUND]	V případě, že je pole vyplněno, ale je nevalidní.
400	FIELD_MISSING	Povinné pole nebylo vyplněno.
400	FIELD_INVALID	Hodnota pole není validní.
400	AC12	[InvalidAccountType] Typ účtu neodpovídá povoleným typům účtů. Např. se jedná o neplatební účet.
400	NARR	[One of the Transactions has already been authorized.]

400	NARR	[Incorrect batch status.]
400	NARR	[Batch payment has already been rejected]

Dávková platba – status (GET/my/batchpayments/{transactionIdentification}/status)

Resource pro zjištění statusu založené dávkové platby.

Charakteristika resource

URI: </batchpayments/{transactionIdentification}/status>

HTTP Metoda: GET

Request URL:

<https://api.kb.cz/sandbox/pisp/v1/batchpayments/{transactionIdentification}/status>

Authorization: request **nevyžaduje** autorizaci uživatele/klienta jako součást volání API

Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne

Třídění: ne

Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“ V tomto poli jsou podporovány pouze znaky bez diakritiky.
X-Request-ID	Text	Ano	Unikátní identifikátor požadavku aplikace TPP.

Path parameter:

PARAMETR	TYP	POVINNÝ	ÚČEL
transactionIdentification	Text	Ano	Identifikátor založené dávky.

Elementy response Dávková platba – Status

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	transactionIdentification	[1..1]	Max35Text	Identifikátor založené dávky.
+	signInfo	[1..1]	±	Informace o stavu a id neautorizované dávky.
++	state	[1..1]	StateCode	Informace o stavu autorizace dávky.
++	signId	[1..1]	Text	Identifikátor autorizačního procesu konkrétní dávky.
+	instructionStatus	[1..1]	StatusCode	Status založené dávky
+	batchDigest	[1..1]	Max50Text	Unikátní ID, zabalené do batch SHA HASH, pomocí kterého jde zkontrolovat, že mezi iniciací a autorizací dávky nedošlo k žádné změně.
+	payments	[0..n]	±	Identifikace konkrétních transakcí.

++	transactionIdentification	[1..1]	Max35Text	Identifikátor založené transakce.
++	instructionIdentification	[0..1]	Max35Text	Identifikace instrukce. Pokud není vyplněna, tak se doplní NOTPROVIDED.
++	instructionStatus	[1..1]	StatusCode	Status založené platby.
++	errorInfo	[0..1]	±	Používá se když je instructionStatus na úrovni transakce "RJCT"
+++	code	[0..1]	Max15Text	Chybový kód
+++	description	[0..1]	Max150Text	Detail popisu chybové zprávy.

Chybové kódy definované pro službu GET status dávkové platby

HTTP STATUS CODE	ERROR CODE	PURPOSE
401	UNAUTHORISED	Chybí kvalifikovaný certifikát.
403	FORBIDDEN	Volání metody, která neodpovídá licenci, nebo neplatný kvalifikovaný certifikát.
404	[ID_NOT_FOUND]	V případě, že je pole vyplněno, ale je nevalidní.
400	FIELD_MISSING	Povinné pole nebylo vyplněno.
400	FIELD_INVALID	Hodnota pole není validní.

10. PSD2 slovník - výběr

API – Application Programming Interface

REST - (Representational State Transfer) - REST je architektura API, která nám umožňuje přistupovat k datům a provádět nad nimi CRUD operace. Architektura, která pracuje obvykle přes protokol HTTP/HTTPS. REST je bezstavový, čímž jednak značně zjednodušuje komunikaci s API a umožňuje paralelní zpracování obsahu. Zároveň ho lze dost snadno použít s HTTP, což je velice rozšířený protokol. V neposlední řadě nám poskytuje určitý standard, takže není problém použít cizí API nebo naopak nabízet vlastní API velkému množství dalších uživatelů. Rozhraní REST je použitelné pro jednotný a snadný přístup ke zdrojům (resources). Zdrojem mohou být data, stejně jako stavy aplikace (pokud je lze popsat konkrétními daty). Všechny zdroje mají vlastní identifikátor URI a REST definuje čtyři základní metody pro přístup k nim (GET, PUT, POST, DELETE). Významy jednotlivých HTTP Verbs jsou následující:

- GET - získání dat
- POST - vytvoření
- PUT - úpravy (upraví celý zdroj - chová se jako SET)
- DELETE - smazání
- PATCH - částečné úpravy

REST API - rozhraní pro distribuované prostředí orientované na data, nikoli na volání procedur jako např. RPC (XML-RPC) či SOAP. Webové služby definují vzdálené procedury a protokol pro jejich volání, REST určuje, jak se přistupuje k datům. REST API využívá metody HTTP jako jsou @GET, @PUT, @POST, @DELETE, @PATCH.

TPP – Third Party Provider (třetí strana registrovaná/licencovaná ČNB)

AIS - Account Information Service

AISP - Account Information Service Provider

PIS - Payment Initiation Service

PISP - Payment Initiation Service Provider

CIS - Card-based Payment Instrument Issuance Service

CISP - Card-based Payment Instrument Issuance Service Provider

ASPSP – Account Servicing Payment Service Provider (banka vedoucí platební účet plátce)