Slovenská záručná a rozvojová banka, a. s.
Štefánikova 27, 811 05 Bratislava – mestská časť Staré Mesto
IČO: 00 682 420, IČ DPH: SK2020804478
zapísaná v obchodnom registri Okresného súdu Bratislava I,
oddiel Sa, vložka č. 3010/B
SWIFT Code: SLZBSKBA, www.szrb.sk

Slovenská záručná a rozvojová banka, a. s.

PSD2 – INFORMÁCIA PRE TRETIE STRANY_v2

História zmien

Verzia dokumentu	Dátum platnosti	Popis zmien v dokumente
API PSD2 v SZRB_v2	19.7.2022	núdzové zmeny: doplnená kapitola 3: JWT token, Príklady http (request / response), zmena portu 98

Obsah

1. Úvod

Pre autorizáciu požiadaviek je použitý autentizačný protokol OAuth 2.0 (popis resouce používaných v rámci tohto protokolu pozrite kapitolu 2.8).

Pre komunikačné rozhranie API sa používa transportný protokol REST (Representational State Transfer). Pre formát zápisu dát dotazu aj odpovede cez API je použitý JSON (JavaScript Object Notation) (výnimkou je formát dát pre požiadavku typu inicializácia platby, kedy je použitý formát XML).

Komunikácia medzi aplikáciou tretej strany a bankou je zabezpečená pomocou SSL protokolu s minimálne 128 bitovým šifrovaním.

Tretia strana bude svoje požiadavky zasielať na vystavené endpointy. Popis jednotlivých metód, ktoré budú pre tretiu stranu k dispozícii, je súčasťou kapitoly 2.9.

1.1 Terminológia

ASPSP - Account Servicing Payment Service Provider – poskytovateľ platobných služieb, v tomto prípade banka.

API PSD2 - komunikačné rozhranie, ktoré umožňuje prostredníctvom súborov programov a funkcií, tretím stranám bezpečne komunikovať v online prostredí s bankou. Cez API rozhranie zadáva jedna aplikácia požiadavku, ktorú druhá aplikácia spracuje. Cez takúto aplikáciu budú môcť klienti napríklad zadávať platby.

TPP - Third Party Provider – tretia strana, subjekt, ktorý sprostredkováva služby banky. Tretia strana môže používať maximálne tri nasledujúce typy služieb (AISP, PISP, PIISP).

Consent - Súhlas klienta s poskytovaním služieb cez sprostredkovateľa – TPP.

AISP - Account Information Service – poskytovateľ služby informovania o platobnom účte - na základe súhlasu klienta poskytuje TPP informácie o platobnom účte a transakciách, ktoré sú vykonané na účte klienta v banke. Napríklad, ak má klient vedené účty vo viacerých bankách, prostredníctvom tretej strany môže vidieť históriu transakcií, prípadne aj zostatky na všetkých týchto účtoch súčasne na jednom mieste (cez aplikáciu alebo portál TPP).

PISP - Payment Initation Service – poskytovateľ služby nepriameho zadania platobného príkazu - ak má TPP povolenú túto službu, môže:

- iniciovať z účtu klienta platbu,

- potvrdiť (autorizovať) odoslanie platby iniciované treťou stranou do banky na spracovanie (ak predtým klient túto platbu autorizoval)
- pýtať sa na stav platby
- overit či má klient na bankovom účte dostatok prostriedkov na zrealizovanie transakcie

PIISP - Payment Instrument Issuer Service Provider – poskytovateľ platobných služieb vydávajúci platobný nástroj (platobnú kartu). TPP si potom bude môcť overiť, či má klient na bankovom účte, ku ktorému TPP vydala kartu, dostatok prostriedkov na zrealizovanie transakcie kartou. Banka odpovie na otázku TPP odpoveďou ÁNO / NIE.

API Gateway – aplikácia v DMZ banky, ktorá poskytuje prístup k PSD2 službám vystavených bankou pre tretie strany.

Internet Banking - Platforma elektronického bankovníctva, prevádzkovaného v SZRB.

PSD2 - Podpora PSD2 funkcionalít.

2. Bezpečnostný model

Základný bezpečnostný model pre prístup k API je založený na kombinácii nižšie uvedených bezpečnostných prvkov (aby TPP mohla posielať požiadavky cez API týkajúce sa účtu / účtov klienta banky, musia byť splnené všetky tieto bezpečnostné prvky):

- Šifrovaná komunikácia medzi TPP a bankou (Použitie platného certifikátu na strane TPP i banky)
- Registrovaný platný záznam TPP v banke (Internet Bankingu)
- Registrovaná aplikácia TPP v banke (s jedinečným client_id a client_secret)
- Existencia platného súhlasu s definovaným prístupom aplikácie TPP k účtom disponenta
- Platný access token (naviazaný na špecifický súhlas, vytvorený disponentom) uvádzaný v hlavičke zaslanej požiadavky cez API

2.1 Šifrovaná komunikácia TPP-Banka

Komunikácia medzi klientskym systémom a bankou predpokladá zabezpečenie pomocou SSL protokolu s minimálne 128 bitovým šifrovaním. TPP musí pre vytvorenie zabezpečeného kanála použiť kvalifikovaný certifikát pre autentizáciu webových serverov podľa eIDAS pre PSD2.

2.2 Registrácia TPP v banke

Ak existuje v databáze Internetbankingu platný záznam TPP, musí TPP cez špecifický endpoint vystaveného PSD2 API vykonať registračné flow v banke.

Pri registračnom flow si TPP v banke zaregistruje svoju aplikáciu / Multibank portál (TPP môže prevádzkovať viac aplikácií). Ak TPP ponúka klientom viac svojich aplikácií, musia každú svoju PSD2 aplikáciu zaregistrovať v banke. TPP dostane ku každej zaregistrovanej aplikácii od banky technické identifikátory (client_id, client_secret). Technické identifikátory sa používajú pri autentizačným flow s použitím OAuth 2.0.

2.3 Súhlas s prístupom TPP k účtom disponenta

Ďalšou podmienkou, ktorá musí byť splnená, aby TPP mohla zasielať požiadavky cez API, je existujúci platný súhlas prístupu aplikácie TPP k účtom disponenta. Súhlas vznikne na základe žiadosti o prístup, ktorú disponent vytvorí na strane banky a autorizuje ju svojím autentizačným zariadením.

Súčasťou uloženého súhlasu sú položky:

- Vybraná aplikácia TPP
- Zoznam oprávnenia k službám (AISP, PISP. PIISP), ktoré povolil disponent pre aplikáciu TPP (v žiadosti sú implicitne povolené všetky služby TPP (okrem služby pre PIISP), ktoré sú evidované v zázname TPP. Služba pre PIISP nie je v žiadosti o súhlas implicitne povolená. Túto službu musí disponent v žiadosti manuálne povoliť.
- Zoznam účtov, ku ktorým disponent povolil prístup (v žiadosti sú implicitne povolené všetky bežné účty, ku ktorým má daný disponent nastavený v banke aktívny prístup (účet sa v súhlase ponúka len vtedy, ak má disponent vo väzbe na klienta, ktorý je majiteľom účtu, povolenú v banke službu PSD2))
- Dátum „Platnosť DO“ vydaného súhlasu

2.4 Popis riešenia

2.4.1 Hlavné vlastnosti API

- API rozhranie: bankové API je riešené ako webová služba (WS).
- API rozhranie:
    - Pre komunikačné rozhranie API je použitý transportný protokol REST (Representational State Transfer).

- Pre formát zápisu dát dotazu aj odpovede cez API je použitý JSON (JavaScript Object notácie) (výnimkou je formát dát pre požiadavku typu inicializácia platby, kedy sa používa formát XML).

- Evidencia aplikácií TPP: TPP si registruje v banke pri registračnom flow 1 .. n aplikácií.
- Požiadavka zaslaná z TPP cez API do banky dostane požadovanú odpoveď iba pri splnení všetkých nasledujúcich podmienok:
  - na základe čísla licencie uvedeného v certifikáte (číslo licencie vrátane prefixu), ktorý TPP používa pri komunikácii je záznam TPP dohľadaný v banke v zoznamu TPP - identické číslo licencie musí byť uvedené v certifikátu aj v zázname TPP v banke,
  - dohľadaný záznam TPP je platný,
  - typ použitej metódy zodpovedá službe (AISP, PISP, PIISP), ktorá je povolená v dohľadanom zázname TPP,
  - access_token použitý v požiadavky je platný
  - na základe použitého access_tokenu (OAuth protokol) uvedeného v požiadavke, je dohľadaný platný súhlas,
  - ak je v tele požiadavky uvedený účet, musí byť tento účet obsiahnutý v súhlase, ktorý bol dohľadaný na základe access tokenu (uvedeného v hlavičke požiadavky),
  - aplikácia TPP má v dohľadanom súhlasu povolenú od disponenta službu (AISP, PISP, PIISP), ktorá zodpovedá metóde použitej v prijatej požiadavke.

2.4.2 Popis workflow – nastavenie prístupu TPP k účtom disponenta

- TPP zaregistruje svoju aplikáciu v banke (cez API vystavené bankou pre TPP) s použitím špecifických metód (viz kapitolu 2.7).
- V okamihu prijatia požiadavky na registráciu aplikácie TPP cez PSD2 API vystavené bankou, prebehne na strane banky v systéme elektronického bankovníctva overenie TPP. Overenie je vykonávané na základe ID licencie vydanej národným regulátorom a certifikátu daného subjektu (ID licencie, uvedené v certifikáte, ktorý TPP používa pri komunikácii cez PSD2 API vystavenej bankou, musí byť obsažené v zázname TPP v banke).
- V prípade, že ID licencie obsiahnuté v certifikáte použitého pri komunikácii TPP cez API, nie je obsiahnuté v žiadnom zázname TPP v databáze IB je postup nasledujúci:
  - TPP kontaktuje pracovníka banky, ktorý vykoná manuálne overenie (TPP banke odovzdá svoj certifikát (bez tajnej časti) s potrebnými dokladmi, na základe ktorých pracovník banky overí a dohľadá danú TPP v databáze IB.

- Po manuálnom overení pracovník banky doplnia do databázy IB do záznamu TPP chýbajúce ID licencie (použité v certifikáte TPP).
- Po doplnení ID licencie bankou, TPP vykoná ďalší pokus registrácie svojej aplikácie cez API.

- Pri registrácii aplikácie TPP pre komunikáciu cez API banky sú po overení TPP v elektronickom bankovníctve vygenerované nasledujúce technické bezpečnostné prvky potrebné pri autentizačným flow s použitím OAuth 2.0:
  - Identifikátor (client_id), ktorý bude TPP pri komunikácii cez API používať
  - secret kód (client_secret), ktorý TPP bude použitý v OAuth protokole pri výmene jednorazového autorizačného kódu za refresh a access token.
- Vygenerované technické bezpečnostné prvky sú odovzdané TPP (TPP tieto technické bezpečnostné prvky dostane pri registrácii cez API ako odpoveď na požiadavku registrácie)
- Od okamihu vygenerovanie technických bezpečnostných prvkov sa názov zaregistrované aplikácie TPP bude ponúkať klientom banky (disponentom) pri vytváraní súhlasov pre prístup TPP k účtom.

2.4.2.1 Postup vytvorenia súhlasu (udelenie prístupu pre TPP) disponentom

- Nasledujú kroky, ktoré musí vykonať disponent.
- Aby mohla TPP zasielať dotazy pres API na účty disponenta alebo vytvárať za disponenta platbu, musí k tomu dať disponent súhlas.
- Disponent klienta banky podpíše zmluvu s TPP.
- Disponent môže súhlas o prístup tretej strany k svojim bežným účtom vytvoriť dvoma spôsobmi:
  - z prostredia Internetbankingu
  - cez Centrálnu autorizačnú stránku, na ktorú je presmerovaný pri aktivácii PSD2 prístupu cez aplikáciu tretej strany.
- Ak je aplikácia tretej strany už zaregistrovaná v banke, disponent klienta banky **sa môže štandardným spôsobom prihlásiť do IB** a v sekcii PSD2 vytvoriť a autorizovať žiadosť o vytvorenie súhlasu pre prístup špecifické aplikácie TPP k svojim bežným účtom. **V žiadosti sú implicitne povolené:**

  - všetky bežné účty, ku ktorým má daný disponent nastavený v banke aktívny prístup (účet sa v súhlase ponúka len vtedy, ak má disponent vo väzbe na klienta, ktorý je majiteľom účtu, povolenú v banke službu PSD2)

  - všetky služby TPP (okrem služby pre PIISP), ktoré sú evidované v zázname TPP registrované v databáze Internet Bankingu.

  Služba pre PIISP nie je v žiadosti o súhlas implicitne povolená. Túto službu musí disponent v žiadosti manuálne povoliť. V prípade, že disponent službu pre PIISP nepovolí pri aktivácii súhlasu, môže ju dodatočne povoliť v internetbankingu editáciou súhlasu.

- Disponent žiadosť o súhlas autorizuje svojim autorizačným zariadením.

- Klient banky (užívateľ aplikácie) si nainštaluje aplikáciu TPP alebo pristupuje k portálu TPP.
- Klient banky (užívateľ aplikácie) si v aplikácii / portáli vyberie svoju banku a spustí registračný workflow.
- Klient je po spustení registračného flow v aplikácii TPP presmerovaný na autentizačný frontend banky (centrálnu autorizačnú stránku) s využitím protokolu OAuth 2.0.
- Klient sa na centrálnej stránke štandardným spôsobom autentizuje (ako v IB). Prihlasovanie PSU - maximálny počet pokusov - 5, maximálny čas bez činnosti - 5 minút.
- **Po autentizácii IB skontroluje, či pre danú TPP aplikáciu a práve prihláseného disponenta existuje platný súhlas (súhlas vytvorený na základe žiadosti z Internet Bankingu alebo z Centrálnej autorizačnej stránky).**
- **Ak platný súhlas nie je dohľadaný**, zobrazí sa stránka, ktorá obsahuje položky žiadosti o súhlas, v ktorej sú implicitne povolené:

  - všetky **bežné** účty, ku ktorým má daný disponent nastavený v banke aktívny prístup

  - všetky služby (okrem služby pre PIISP), ktoré sú evidované v zázname tretej strany registrované v databáze Internet Bankingu.

  Služba pre PIISP nie je v žiadosti o súhlas implicitne povolená. Túto službu musí disponent v žiadosti manuálne povoliť. V prípade, že disponent službu pre PIISP nepovolí pri aktivácii súhlasu, môže ju dodatočne povoliť v internetbankingu editáciou súhlasu.

  - Disponent žiadosť o súhlas autorizuje svojim autorizačným zariadením.

- **Ak v IB pre danú aplikáciu TPP existuje platný súhlas vytvorený disponentom**, v odpovedi protokole OAuth získa TPP jednorazový autorizačný kód. TPP následne kontaktuje endpoint token vystavený na frontende banky, aby tento jednorazový autorizačný kód vymenil za dvojicu tokenov access a refresh token.
- Aplikácia TPP následne Access token používa pri komunikácii s PSD2 API vystavený bankou. Vnútrobankovné systémy následne budú žiadať o overenie platnosti tokenu a získanie príslušného scope (AIS / PIS / PIIS) a príslušné užívateľské identity, ku ktorej token patrí.

2.4.3 E-Banking

2.4.3.1 Prehľad „Prehľad súhlasov“

Ak používateľ vyberie ponuku "Prehľad súhlasov ", zobrazí sa prehľad všetkých súhlasov, ktoré vznikli na základe žiadostí o udelenie prístupu pre TPP, ktorý prihlásený používateľ vytvoril.

Štruktúra prehľadu - prehľad obsahuje nasledujúce stĺpce:

Názov stĺpca	Popis
Tretia strana	Názov tretej strany, na ktorú je registrovaná aplikácia TPP, ktorú disponent uviedol v žiadosti

Názov stĺpca	Popis
Aplikácia tretej strany	Názov aplikácie tretej strany, ktorú disponent uviedol v žiadosti
Povolené služby	Množina služieb (AISP, PISP, PIISP), ktoré disponent povolil na užívanie pre tretiu stranu
Platnosť do	Dátum ukončenia platnosti súhlasu. V prípade, že si platnosť súhlasu užívateľ nastavil pri vytváraní žiadosti o súhlas, je tento dátum uvedený v súhlase ihneď po vytvorení súhlasu. Pokiaľ nebude platnosť súhlasu do žiadosti vyplnená, bude vytvorený súhlas platný do jeho manuálneho zrušenia disponentom.

Na začiatku každého zobrazeného riadku je grafický prvok, pomocou ktorého si užívateľ IB otvorí detail súhlasu.

2.4.3.2 Detail „Súhlas s prístupom pre tretiu stranu“

Názov položky	Popis
Meno tretej strany	Meno TPP, na ktorú je registrovaná aplikácia TPP, pre ktorú disponent povolil prístup k svojim účtom.
Aplikácia tretej strany	Meno aplikácie TPP, pre ktorú disponent povolil prístup k svojim účtom.
Služby	Množina služieb, ktoré disponent nastavil do žiadosti o povolenie prístupu pre TPP. V súhlase sa zobrazujú služby: • AISP (Zoznam účtov povolených pre TPP, Otázka na zostatok účtu, Prehľad transakcií) • PISP (Vytvorenie platby, Stav spracovania príkazu v banke, Autorizácia platby, Overenie dostatočných prostriedkov na účte, Zrušenie platby) • PIISP (Overenie dostatočných prostriedkov na účte)
Účty	Zoznam účtov, ktoré disponent v žiadosti o povolenie prístupu pre TPP povolil.
Platnosť do	Dátum a čas ukončenia platnosti súhlasu.

2.4.3.2.1 Ovládacie prvky v detaile AKTÍVNEHO SÚHLASU

V detaile každého **aktívneho** súhlasu sa užívateľovi ponúkajú tlačidlá:

Tlačidlo	Popis
Žiadosť o ukončenie	po stlačení tlačidla sa vytvorí žiadosť o ukončenie platnosti súhlasu (do záznamu súhlasu sa po autorizácii žiadosti do položky "Platnosť do" uloží dátum a čas, kedy bola autorizácia žiadosti o ukončení súhlasu vykonaná)
Žiadosť o zmenu	po stlačení tlačidla sa vytvorí nová žiadosť o udelenie prístupu, ktorá v sebe bude mať predvyplnené údaje súhlasu, z ktorého je zmena vykonávaná.

2.4.3.2.2 Ovládací prvky v detaile UKONČENÉHO SÚHLASU

V detaile každého **ukončeného** súhlasu sa užívateľovi ponúka tlačidlo:

Tlačidlo	Popis
Žiadosť o obnovenie	po stlačení tlačidla sa vytvorí žiadosť o obnovenie ukončeného súhlasu. Vytvorená žiadosť v sebe má predvyplnené údaje súhlasu, z ktorého je vykonávaná.

2.5 Služby podporované v API PSD2

- riešenie PSD2 umožňuje tretej strane používať cez WS služby PSD2 popísané v nasledujúcich podkapitolách.

2.5.1 Metódy pre automatickú registráciu aplikácie TPP cez API

Metóda služby	Popis
Registrácia aplikácie TPP (JSON)	prostredníctvom tejto služby TPP s platným certifikátom a licenčným číslom vykoná automatickú registráciu svojej aplikácie v banke a v odpovedi dostane k registrované aplikácii technické bezpečnostné prvky (client_id a client_secret)
Zmena registrácie aplikácie (JSON)	prostredníctvom tejto služby bude TPP môcť vykonať zmenu registračných údajov
Zrušenie registrácie aplikácie (JSON)	prostredníctvom tejto služby bude TPP môcť zrušiť registráciu aplikácie

2.5.2 Metódy oblasti AISP

Metóda služby	Popis
AccountInformation (JSON)	prostredníctvom tejto služby dostane disponentom autorizovaná tretia strana prehľad zostatkov bankového účtu disponenta vedeného v banke
AccountTransaction (JSON)	prostredníctvom tejto služby dostane disponentom autorizovaná tretia strana prehľad transakcií
AccountList (JSON)	služba na požiadavku vráti zoznam účtov, ktoré disponent uviedol v súhlase na používanie s konkrétnym TPP (nie zoznam všetkých klientskych účtov) bez zostatkov

2.5.3 Metódy oblasti PISP

Metóda služby	Popis
Standard payment initialization (XML)	prostredníctvom tejto služby disponentom autorizovaná tretia strana iniciuje (vytvorí) jeden SEPA príkaz z bankového účtu disponenta vo formáte XML (PAIN.001). TPP následne použije na iniciovaný príkaz službu "/Authorize" - klient banky je presmerovaný na centrálnu autentizačnú stránku banky a tu daný príkaz autorizuje svojim autentizačným zariadením.
Payment status (JSON)	zisťovanie stavu platobného príkazu
Standard payment submission (JSON)	autorizácia platby treťou stranou, (daná platba musí byť iniciovaná touto treťou stranou)
Balance check (JSON)	overenie dostatočného zostatku na účte
Cancel payment (JSON)	zrušenie platby, ktorá ešte nebola autorizovaná treťou stranou (tretia strana nepoužila metódu "Standard payment submission") a ktorá bola vytvorená prostredníctvom služby PISP Standard payment initialization (XML)

2.5.4 Metódy oblasti PIISP

Metóda služby	Popis
Balance check (JSON)	overenie dostatočného zostatku na účte

2.6 Endpointy použité pre API PSD2

Banka v rámci riešenia PSD2 prevádzkuje dve prostredia:

- Testovacie prostredie - určené pre vývojárov TPP; na tomto prostredí TPP môže vykonávať testy svojich vyvinutých aplikácií.
- Produkčné prostredie - TPP sa môže pripojiť k tomuto prostrediu po otestovaní svojej aplikácie na testovacom prostredí.

Prístup do požadovaného prostredia (testovacie alebo produkčné) je TPP umožnený na základe workflow opísaného v kapitole 2.4.2.

Každé prostredie je prevádzkované na konkrétnej Root URL adrese viď nasledujúca tabuľka.

Prostredie	Root URL	Popis
Produkčné prostredie	https://api.szrb.sk:98	V dokumente je táto root URL adresa primárne uvádzaná vo všetkých endpointoch.
Testovacie prostredie	https://apitest.szrb.sk:98	Ak chce TPP použiť testovacie PSD2 API dostupné na strane banky, použije sa v popisovaných Endpointoch namiesto koreňovej adresy **https://api.szrb.sk:98** adresa pre testovacie prostredie (**https://apitest.szrb.sk:98**)

2.6.1 Endpointy pre OAuth (autorizácia klienta, autorizácia platby klientom, vydávanie tokenov)

Endpoint	Typ metódy	Popis
https://api.szrb.sk:98/auth/oauth/authorize	GET	Endpoint používaný v rámci Oauth **Authorization code grant pre:** - **autorizáciu klienta,** - **autorizáciu platby klientom**.
https://api.szrb.sk:98/auth/oauth/token	POST	Endpoint používaný v rámci Oauth **Authorization code grant pre:** - **vygenerovanie novej dvojice access_token a refresh_token** - **obnovenie access tokenu** - **vygenerovanie jednorazového access tokenu pre použitie v metóde /api/v1/payment/submission**

2.6.2 Endpointy pre PSD2 API (registračné resource (Enrollment), volanie metód AISP, PISP, PIISP)

2.6.2.1 Enrollment (registrácia aplikácie v banke)

Endpoint	Typ metody	Popis
https://api.szrb.sk:98/api/enroll	POST	Endpoint pre metódu, prostredníctvom ktorej TPP s platným certifikátom a licenčným číslom vykoná automatickú registráciu svojej aplikácie v banke a v odpovedi dostane k registrované aplikácii technické bezpečnostné prvky (client_id a client_secret).
https://api.szrb.sk:98/api/enroll/{client_id}	PUT	Zavolaním tohto resource môže TPP požiadať o zmenu registračných údajov pre konkrétnu aplikáciu.
https://api.szrb.sk:98/api/enroll/{client_id}	DELETE	Zavolaním tohto resource môže TPP požiadať o zrušenie registrácie konkrétnej aplikácie.
https://api.szrb.sk:98/api/enroll/{client_id}/renewSecret	POST	Zavolaním tohto resource môže TPP požiadať o vydanie nového client_secret k danej aplikáci.

2.6.2.2 AISP

Endpoint	Typ metódy	Popis
https://api.szrb.sk:98/api/v1/accounts/information	POST	Endpoint pre metódu, prostredníctvom ktorej dostane disponentom autorizovaná tretia strana prehľad zostatkov bankového účtu disponenta vedeného v danej banke
https://api.szrb.sk:98/api/v1/accounts/transactions	POST	Endpoint pre metódu, prostredníctvom ktorej dostane disponentom autorizovaná tretia strana prehľad transakcií
https://api.szrb.sk:98/api/v2/accounts	GET	Endpoint pre metódu, ktorá na požiadanie vráti zoznam účtov, ktoré disponent uviedol v súhlasu k používaniu s konkrétnym TPP (nie zoznam všetkých účtov disponenta) bez zostatkov

2.6.2.3 PISP

Endpoint	Typ metódy	Popis
https://api.szrb.sk:98/api/v1/payments/standard/iso	POST	Endpoint pre metódu **Standard payment initialization (XML)** - disponentom autorizovaná tretia strana iniciuje (vytvorí) jeden SEPA príkaz z bankového účtu disponenta.
https://api.szrb.sk:98/api/v1/payments/submission	POST	Endpoint pre metódu **Standard payment submission –** autorizácia platby tretou stranou (platbu predtým musí povoliť svojou autorizáciou disponent).
https://api.szrb.sk:98/api/v1/payments/{orderId}/status	GET	Endpoint pre metódu **Payment order status –** zisťovanie stavu platobného príkazu
https://api.szrb.sk:98/api/v1/payments/{orderId}/rcp	DELETE	Endpoint pre metódu **Cancel payment –** zrušenie platby, ktorá ešte nebola autorizovaná treťou stranou (tretia strana nepoužila metódu "Standard payment submission") a ktorá bola vytvorená prostredníctvom služby PISP Standard payment initialization (XML)
https://api.szrb.sk:98/api/v1/accounts/balanceCheck	POST	Endpoint pre metódu **Balance check** - overenie, či má klient na bankovom účte dostatok prostriedkov na zrealizovanie transakcie

2.6.2.4 PIISP

Endpoint	Typ metódy	Popis
https://api.szrb.sk:98/api/v1/accounts/balanceCheck	POST	Endpoint pre metódu **Balance check** - overenie, či má klient na bankovom účte, ku ktorému TPP vydala kartu, dostatok prostriedkov k zrealizovaniu transakcie kartou

2.7 Registračné resource vystavené bankou (Enrollment)

Nasledujúce kapitoly popisujú metódy, pomocou ktorých TPP žiada o registráciu svojej aplikácie v banke, prípadne môže vykonať zmeny alebo zrušenia registrácie svojej aplikácie.

2.7.1 Automatické generovanie technických identifikátorov

Pre zavolanie resource je potreba:

- **Použiť platný certifikát**

Výstupom sú parametre client_id a client_secret, ktoré TPP potrebuje pre následné získanie dvojice tokenov access_token a refresh_token.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api/**enroll**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api/**enroll**

Request			
Atribut	Povinný	Typ	Popis
redirect_uris	Áno	Array of strings e.g. URL [Max 3x 2047 B]	Zoznam URL kam môže byť flow autentizácie na konci presmerované. Autorizačný request musí obsahovať práve jedno z týchto registrovaných URI v presnom formáte.
client_name	Áno	String [Max 255 B]	Meno TPP aplikácie
client_name#en-US	Nie	String [Max 1024 B]	Meno TPP aplikácie v príslušnom jazyku / kódovanie.
client_type	Áno	String	OAuth definuje dva typy klientov (Confidential / Public). ASPSP (banka) podporuje len typ Confidental.
logo_uri	Nie	URI [Max 2047 B]	URI loga aplikácie (resp. Miesto odkiaľ je možné ho pri registrácii stiahnuť)
contacts	Áno	Array of strings e-mail [Max 10x 255 B]	Zoznam E-mail adries, kontakty na zodpovednú osobu na strane TPP aplikácie.
scopes	Nie	Array of strings [Max 10x 255 B]	Pole požadovaných Scopes pre aplikáciu. Pri registrácii sú Scopes validované proti obsahu použitého certifikátu a proti Scopes uvedených v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.
licence_number	Áno	String [Max 1024 B]	Licenčné číslo, ktoré má TPP pridelené od národného regulátora. Licenčné číslo je validované proti licenčnému číslu uvedenému v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.

Response			
Atribut	Povinný	Typ	Popis
client_id	Áno	String	client_id priradené aplikácii. Toto ID je používané pri spustení autentizačného procesu a pri komunikačnom procese (výmene jednorazového code za dvojicu tokenov access_token a refresh_token a pri obnovení tokenu).
client_secret	Áno	String	Client_secret - password / token vydený bankou (ASPSP) pre TPP aplikácii (client_id)
client_secret_expires_at	Nie	DateTime	Defaultná hodnota je 0 (client_secret nikdy neexspiruje). V opačnom prípade je uvedená hodnota v sekundách od dátumu 1970-01-01T0:0:0Z
api_key	Nie	String	API kľúč, ktorý aplikácia používa pri komunikácii s API banky. **API kľúč nie je v tomto riešení bankou podporovaný** (v odpovedi v položke uvedené "NOT_PROVIDED")
redirect_uris	Áno	Array of strings e.g. URL [Max 3x 2047 B]	Zoznam URL kam môže byť flow autentizácie na konci presmerované. Autorizačný request musí obsahovať práve jedno z týchto registrovaných URI v presnom formáte.
client_name	Áno	String [Max 255 B]	Meno TPP aplikácie
client_name#en-US	Nie	String [Max 1024 B]	Meno TPP aplikácie v príslušnom jazyku / kódovanie.
client_type	Áno	String	OAuth definuje dva typy klientov (Confidential / Public). **ASPSP (banka) podporuje len typ Confidental.**
logo_uri	Nie	URI [Max 2047 B]	URI loga aplikácie (resp. Miesto odkiaľ je možné ho pri registrácii stiahnuť)
contacts	Áno	Array of strings e-mail [Max 10x 255 B]	Zoznam E-mail adries, kontakty na zodpovednú osobu na strane TPP aplikácie.
scopes	Nie	Array of strings [Max 10x 255 B]	Pole požadovaných Scopes pre aplikáciu. Pri registrácii sú Scopes validované proti obsahu použitého certifikátu a proti Scopes uvedených v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.

Response			
licence_number	Áno	String [Max 1024 B]	Licenčné číslo, ktoré má TPP pridelené od národného regulátora. Licenčné číslo je validované proti licenčnému číslu uvedenému v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.

Chybové kódy		
HTTP Status	Error kód	Popis
400	**invalid_request**	Nevalidný request. V dotazu chýba povinné pole alebo je v nevhodnom / nevalidním formáte.
400	**invalid_scope**	Nevalidný scope v požiadavke.
400	**invalid_redirect_uri**	Hodnota jedného alebo viacerých redirect uri nie je validný
401	**invalid_client**	Nevalidný client_id.
401	**unauthorized_client**	TPP nie je oprávnený vykonávať tento dotaz.
401	**access_denied**	Autorizačný server odmietol prístup.
403	**insufficient_scope**	Napr. nedostatočné oprávnenia pre použitie požadovaného scope
500, 503	**server_error**	Chyba autorizačného servera.

Príklad použitia viď zdroj kapitola 3.2.1.1.

2.7.2 Zmena registračných údajov

Zavolaním tohto resource môže TPP požiadať o zmenu registračných údajov pre konkrétnu aplikáciu.

Pre zavolanie resource je potreba:

- Použiť platný certifikát

- Použiť client_id, vydané k tomuto TPP.

Výstupom je prehľad zmenených údajov.

Testovacie prostredie

Endpoint: PUT https://apitest.szrb.sk:98/api**/enroll/{client_id}**

Produkčné prostredie

Endpoint: PUT https://api.szrb.sk:98/api**/enroll/{client_id}**

Request			
Atribut	Povinný	Typ	Popis
redirect_uris	Áno	Array of strings e.g. URL [Max 3x 2047 B]	Zoznam URL kam môže byť flow autentizácie na konci presmerované. Autorizačný request musí obsahovať práve jedno z týchto registrovaných URI v presnom formáte.
client_name	Áno	String [Max 255 B]	Meno TPP aplikácie
client_name#en-US	Nie	String [Max 1024 B]	Meno TPP aplikácie v príslušnom jazyku / kódovanie.
client_type	Áno	String	OAuth definuje dva typy klientov (Confidential / Public). **ASPSP (banka) podporuje len typ Confidental.**
logo_uri	Nie	URI [Max 2047 B]	URI loga aplikácie (resp. Miesto odkiaľ je možné ho pri registrácii stiahnuť)
contacts	Áno	Array of strings e-mail [Max 10x 255 B]	Zoznam E-mail adries, kontakty na zodpovednú osobu na strane TPP aplikácie.
scopes	Nie	Array of strings [Max 10x 255 B]	Pole požadovaných Scopes pre aplikáciu. Pri registrácii sú Scopes validované proti obsahu použitého certifikátu a proti Scopes uvedených v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.

Response			
Atribut	Povinný	Typ	Popis
client_id	Áno	String	client_id priradené aplikácii bankou.
client_secret_expires_at	Nie	DateTime	Defaultná hodnota je 0 (client_id nikdy neexspirujú). V opačnom prípade je uvedená hodnota v sekundách od dátumu 1970-01-01T0:0:0Z
redirect_uris	Áno	Array of strings e.g. URL [Max 3x 2047 B]	Zoznam URL kam môže byť flow autentizácie na konci presmerované. Autorizačný request musí obsahovať práve jedno z týchto registrovaných URI v presnom formáte.
client_name	Áno	String [Max 255 B]	Meno TPP aplikácie
client_name#en-US	Nie	String [Max 1024 B]	Meno TPP aplikácie v príslušnom jazyku / kódovanie.
client_type	Áno	String	OAuth definuje dva typy klientov (Confidential / Public). **ASPSP (banka) podporuje len typ Confidental.**

Response			
logo_uri	Nie	URI [Max 2047 B]	URI loga aplikácie (resp. Miesto odkiaľ je možné ho pri registrácii stiahnuť)
contacts	Áno	Array of strings e-mail [Max 10x 255 B]	Zoznam E-mail adries, kontakty na zodpovednú osobu na strane TPP aplikácie.
scopes	Nie	Array of strings [Max 10x 255 B]	Pole požadovaných Scopes pre aplikáciu. Pri registrácii sú Scopes validované proti obsahu použitého certifikátu a proti Scopes uvedených v zázname TPP, ktorý v tom čase už musí existovať v databáze IB.

Chybové kódy		
HTTP Status	Error kód	Popis
400	**invalid_request**	Nevalidný request. V dotazu chýba povinné pole alebo je v nevhodnom / nevalidním formáte.
400	**invalid_scope**	Nevalidný scope v požiadavke.
400	**invalid_redirect_uri**	Hodnota jedného alebo viacerých redirect uri nie je validný
401	**invalid_client**	Nevalidný client_id.
401	**unauthorized_client**	TPP nie je oprávnený vykonávať tento dotaz.
401	**access_denied**	Autorizačný server odmietol prístup.
403	**insufficient_scope**	Napr. nedostatočné oprávnenia pre použitie požadovaného scope
500, 503	**server_error**	Chyba autorizačného servera.

Příklad použití viz kapitola 3.2.1.2.

2.7.3 Zmazanie aplikácie

Zavolaním tohto resource môže TPP požiadať o zmazanie údajov a prístupu konkrétnu aplikáciu.

Pre zavolanie resource je potreba:

- Použiť platný certifikát

- Použiť platné client_id, ktoré je vydané tomuto TPP.

Výstupom je potvrdenie o zmazaniu.

Testovacie prostredie

Endpoint: DELETE https://apitest.szrb.sk:98/api/**enroll/{client_id}**

Produkčné prostredie

Endpoint: DELETE https://api.szrb.sk:98/api/**enroll/{client_id}**

Ak sa zmazanie aplikácia vykoná, je vrátená odpoveď HTTP 204 ako úspešná odozva na zmazanie záznamu aplikácie s konkrétnym client_id).

Chybové kódy		
HTTP Status	Error kód	Popis
400	**invalid_request**	Nevalidný request. V dotaze chýba povinné pole alebo je v nevhodnom / nevalidnom formáte.
401	**invalid_client**	Nevalidný client_id.
401	**unauthorized_client**	TPP nie je oprávnený vykonávať tento dotaz.
401	**access_denied**	Autorizačný server odmietol prístup.
500, 503	**server_error**	Chyba autorizačného servera.

Príklad použití viz kapitola 3.2.1.3.

2.7.4 Žiadosť o nový client_secret

Zavolaním tohto resource môže TPP požiadať o vydanie nového client_secret.

Pre zavolanie resource je potreba použiť:

- Platný certifikát

- Platný client_id, ktoré je vydané tomuto TPP.

Pôvodný client_secret bude týmto pořiadavkom zrušený.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api/**enroll/{client_id}/renewSecret**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api/**enroll/{client_id}/renewSecret**

Response			
Atribut	Povinný	Typ	Popis
client_id	Áno	String	client_id priradené aplikácii.
client_secret	Áno	String	Client_secret - password / token vydený bankou (ASPSP) pre TPP aplikácii (client_id)

Response			
client_secret_expires_at	Nie	DateTime	Defaultná hodnota je 0 (client_id nikdy neexspirujú). V opačnom prípade je uvedená hodnota v sekundách od dátumu 1970-01-01T0:0:0Z

Chybové kódy		
HTTP Status	Error kód	Popis
400	**invalid_request**	Nevalidný request. V dotazu chýba povinné pole alebo je v nevhodnom / nevalidním formáte.
401	**invalid_client**	Nevalidný client_id.
401	**unauthorized_client**	TPP nie je oprávnený vykonávať tento dotaz.
401	**access_denied**	Autorizačný server odmietol prístup.
500, 503	**server_error**	Chyba autorizačného servera.

Příklad použití viz kapitola 3.2.1.4.

2.8 Autentizácia a Autorizácia requestu (OAuth2)

Autorizácia requestu je založená na autorizačnom flow koncepte OAuth2 zabezpečeného tokenom - aplikácia len skontroluje platnosť tokenu použitého v hlavičke požiadavky, ktorý TPP poskytuje pre každé volanie ako dôkaz, že môže pristupovať k požadovaným údajom.
V rámci API je autorizačný token považovaný za krátkodobý a bezstavový prvok, ktorý musí byť použitý v každom volaní API, ktoré požaduje autorizáciu requestu.

Základom riešenia je použitie OAuth2 otvoreného protokolu pre vystavovanie autorizačných tokenov – **je podporovaný iba autorizačný framework Authorization code grant**.

2.8.1 OAuth2 Authorization Code Grant

V rámci protokolu OAuth2 sa v prípade autorizačného frameworku Authorization code grant jedná o spôsob, ako partnerské aplikácii vydať refresh token aj access token ako výsledok identifikácie a autentizácie užívateľa. Krátkodobý access token partnerská aplikácia používa pre komunikáciu s API banky a po jeho expirácii môže použiť refresh token pre vyžiadanie nového access tokenu.

2.8.1.1 Základné vlastnosti

- Access token je vydávaný ako krátkodobý (3600 s)

- Access token je vydávaný pre konkrétnu aplikáciu a konkrétneho užívateľa, pre inú aplikáciu a užívateľa ho nie je možné úspešne použiť
- Refresh token nie je možné priamo použiť pre komunikáciu s API, má dlhú platnosť (v prípade PSD2 90 dní)
- Banka a TPP aplikácia spolu zdieľa spoločné "tajomstvo" (client secret)
- Výsledkom identifikácie a autentizácie užívateľa je jednorazový code, ktorý aplikácia tretej strany môže s použitím **client_id** a **client secret** vymeniť za refresh token a access token
- Samotný jednorazový code bez znalosti client secret nie je možné použiť

2.8.1.2 Popis Code grand flow

Podmienky použitia flow:

- Aplikácia TPP má od banky pridelené vlastné jedinečné client_id a TPP backend server pozná pre dané client_id aj client secret
- Pri vydaní client_id a client_secret banka získa informáciu o redirect uri - teda o URL, kam má presmerovať užívateľa po úspešnej autentifikácii

Jednotlivé kroky code grant flow:

1. TPP zavolá resource banky /Authorize a následne je užívateľ (klient banky) presmerovaný na centrálnu autentizačnú stránku pre vykonanie identifikácie a autentizácie užívateľa (klienta banky).
2. Prebieha identifikácia a autentizácia klienta - tieto kroky sú plne v réžii banky
3. Po úspešnej autentizácii banka vygeneruje code a presmeruje s ním používateľa na URI, ktoré bolo súčasťou požiadavky /Authorize (redirect_uri)
4. TPP použije resource /token na získanie refresh_tokenu a access_tokenu. Pri volaní tohto zdroja TPP použije:
   - v hlavičke v položke Authorization dvojicu client_id a client_secret, ktorá však musí byť zašifrovaná pomocou Base64 (formát použitého reťazca: Basic <hodnota vygenerovaná pomocou Base64 (client_id:client_secret)>)
   - a v tele požiadavky hodnotu code, ktorý dostala v odpovedi predchádzajúcej požiadavky / Authorize.
5. Aplikácia TPP používa pri komunikácii na API banky v prípadoch, keď je to potrebné, v hlavičke požiadavky, získaný access_token
6. Banka interne vykonáva overovanie access_tokenu. Pri tomto overenie získava identitu používateľa, na základe ktorého autentizácie bol access token vydaný.

2.8.1.3 Autorizačný resource

Ak neexistuje platná dvojica tokenov (access_token a refresh_token), musí TPP vytvoriť Autorizačný požiadavku, na základe ktorej je klient banky z aplikácie presmerovaný na centrálnu autentizačný stránku banky, kde danú požiadavku následne autorizuje. Požiadavka je typu **OAuth 2.0 Authorization Code Grant s PKCE rozšírením**.

Testovacie prostredie

Endpoint: GET https://apitest.szrb.sk:98/auth/oauth/**authorize**

Produkčné prostredie

Endpoint: GET https://api.szrb.sk:98/auth/oauth/**authorize**

Request			
Atribut	Povinný	Typ / hodnota	Popis
response_type	Áno	**code**	Povinný parameter. Hodnotou parametra je určené, aký typ autentizačného flow je požadovaný. V tomto prípade sa jedná o code grant. Pre autentizačný proces to znamená, že výsledkom tejto požiadavky bude jednorazový auth_code, ktorý TPP následne pomocou ďalšej požiadavky (metódou token) zamení za dvojicu tokenov access_token a refresh_token
client_id	Áno	String	Jedinečný identifikátor, ktorý banka vygenerovala pre aplikáciu TPP
redirect_uri	Áno	URL	URL kam je na konci presmerované flow autentizácie. Toto URL je stanovené už pri vydaní client_id a v rámci autentizácia je tento parameter validovaný proti URL zavedenému k client_id v zázname aplikácie registrované v banke. Hodnota by sa mala zhodovať s jednou z hodnôt uvedených v zázname registrované aplikácie.
Scope	Áno	String	Jedná sa o pole aplikácií požadovaných scope (oprávnenie). V prípade PSD2 to môžu byť role AISP, PISP, PIISP. Napr. ak je TPP držiteľom viac oprávnenia, môže tu pre svoju aplikáciu požiadať len o jedno z nich alebo viac. Ak je použitých viac typov scope, sú oddelené medzerou.
state	Áno	Libovolný string [min 128 bits]	Parametrom sa zvyšuje bezpečnosť komunikácie pri presmerovaní. Chráni pred útokmi CSRF a odovzdáva informácie z aplikácie prostredníctvom toku autentizácie.
code_challenge	Áno	String	code_challenge = BASE64URL-ENCODE(SHA256(ASCII(code_verifier))) viz. zdroj [3] RFC 7636 (OAuth PKCE)
code_challenge_method	Áno	String	S256

Response			
Atribut	Povinný	Typ	Popis
Code	Áno	String	Jednorazový autorizačný kód
State	Áno	String	Hodnota atribútu odovzdaného z TPP požiadavky

Chybové kódy

- Chybové kódy sú definovány podľa [1] RFC 6749, kapitola 4.1.2.1

Príklad použitia viz kapitolu 3.2.2.1.

2.8.1.4 Získanie tokenov

Ak TPP na základe odpovede požiadavky /Authorize dostane autorizačný kód (code) a string uvedený v položke state je validný (hodnota state je v odpovedi zhodná s hodnotou state, ktorá bola uvedená v požiadavke), môže TPP požiadať o prístupové tokeny z ASPSP pomocou autorizačného kódu. TPP zašle spoločne s týmto autorizačným kódom (ktorý musí byť uvedený v tele požiadavky) aj client_id a client_secret (ktoré však musí byť uvedené v hlavičke požiadavky zakódované pomocou Base64).

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/auth/oauth/**token**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/auth/oauth/**token**

Request			
Atribut	Povinný	Typ	Popis
code	Áno	string	Autorizačný code navrátený z autentizačného flow (code grant)
redirect_uri	Áno	URL	URL redirect zhodné s URL doručenom v autentizačnom requestu
grant_type	Áno	**authorization_code**	Existujúca definícia / zvyklosti OAuth2 bude táto hodnota authorization_code, ak dochádza k výmene code za dvojicu tokenov access_token a refresh_token.
code_verifier	Áno	String	code_verifier slúži na generovanie code_challenge z predchádzajúcej žiadosti o minimálnej dĺžke 43 znakov a maximálnou dĺžkou 128 znakov

Response			
Atribut	Povinný	Typ	Popis
access_token	Áno	string	Krátkodobý (v niektorých prípadoch jednorazový) token (platnosť tokenu je 3600s), ktorý je možné znovu vygenerovať použitím refresh_tokenu. Tento token slúži na autorizáciu requestu na API.
expires_in	Áno	number	Zostávajúci čas do expirácie access_tokenu - v sekundách.
refresh_token	Áno	String	Dlhodobý token (platnosť 90 dní) vydaný na základe výmeny za jednorazový **code**.
token_type	Áno	String	Typ tokenu “Bearer”
scope	Nie	String	Zoznam Scope oddelených medzerou, pre ktorých je token vydaný.

Chybové kódy

- Chybové kódy sú definované podľa [1] RFC 6749, kapitola 5.2

Príklad použitia viz kapitola 3.2.2.2.

2.8.1.5 Obnovenie Access tokenu

TPP môže po expirácii access_tokenu prostredníctvom refresh tokenu požiadať o nový. Pre to je možné použiť resouce "Získanie tokenov" s nižšie uvedenými parametrami. TPP zašle spoločne s refresh_token (ktorý musí byť uvedený v tele požiadavky) aj client_id a client_secret (ktoré však musí byť uvedené v hlavičke požiadavke zakódované pomocou Base64).

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/auth/oauth**/token**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/auth/oauth**/token**

Request			
Atribut	Povinný	Typ	Popis
grant_type	Ano	**refresh_token**	Existujúca definícia / zvyklosti OAuth2 bude táto hodnota refresh_token, ak dochádza k obnoveniu access_tokenu na základe refresh_token.
refresh_token	Ano	String	Validný **refresh_token**

Request			
scope	Ano	String	Rozsah scope o prístup. Ak sa používa rozsah, je skontrolovaný proti scope uvádzaným v zázname TPP, aplikáciu TPP a súhlasu, ktorý nastavil užívateľ aplikácie.

Response			
Atribut	Povinný	Typ	Popis
access_token	Ano	string	Krátkodobý (v niektorých prípadoch jednorazový token) token (platnosť tokenu je 3600s), ktorý je možné znovu vygenerovať použitím refresh_tokenu. Tento token slúži na autorizáciu requestu na API.
token_type	Ano	String	Typ tokenu “Bearer”
expires_in	Ano	number	Zostávajúci čas do expirácie access_tokenu - v sekundách.
refresh_token	Ano	String	Dlhodobý token (platnosť 90 dní).

Chybové kódy

- Chybové kódy sú definované podľa [1] RFC 6749, kapitola 5.2

Príklad použitia viz kapitola 3.2.2.3.

2.9 Popis metód používaných pre poskytovateľov služieb (TPP)

2.9.1 Všeobecná definícia hlavičky požiadavky

Štruktúra hlavičiek uvedených v tejto kapitole sa používa u všetkých nižšie uvedených metód služieb pre AISP, PISP, PIISP.

Hlavička pre Request

Attribute	Mandatory	Typ	Popis
Host	Ano	String	Doménové meno servera a číslo portu
Content-Type	Ano	String	application/json alebo application/xml
Authorization	Ano	String	Typ autorizácie definovaný podľa RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage
Request-ID	Ano	String	Jedinečný identifikátor konkrétnej požiadavky. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).

Attribute	Mandatory	Typ	Popis
Correlation-ID	Ne	String	Jedinečný korelačný identifikátor, možno ho použiť ako kontrolu prepojenie požiadavky a odpovede. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).
Process-ID	Ne	String	Identifikátor obchodného alebo technického procesu, na základe ktorého možno párovať sadu dvojíc požiadaviek a odpovedí. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).
PSU–IP-Address	Ano	String	IP adresa zákazníka (disponenta banky), z ktorej je pripojený k infraštruktúre TPP.
PSU-Device-OS	Ano	String	Identifikácia zariadenia zákazníka (disponenta banky), alebo operačného systému, z ktorého je pripojený k infraštruktúre TPP
PSU-User-Agent	Ano	String	Identifikácia webového prehliadača zákazníka alebo identifikácia klientskeho zariadenia, z ktorého je pripojený k infraštruktúre TPP
PSU-Geo-Location	Ne	String	Súradnice GPS aktuálnej polohy zákazníka v okamihu pripojenia k infraštruktúre TPP. (Požadovaný formát GPS: zemepisná šírka, zemepisná dĺžka)
PSU-Last-Logged-Time	Ne	DateTime	Dátum a čas, kedy bol používateľ prihlásený k aplikácii TPP (formát RFC3339)
PSU-Presence	Ne	Enum	Stav prítomnosti používateľa (PSU) počas volania na API. Hodnota parametra môže byť "true" (PSU je prítomný) alebo "false" (PSU nie je prítomný).

Hlavička pre Response

Attribute	Mandatory	Typ	Popis
Content-Type	Ano	String	application/json nebo application/xml
Response-ID	Ano	String	Jedinečný identifikátor konkrétnej odpovede. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).
Correlation-ID	Ne	String	Jedinečný korelačný identifikátor, možno ho použiť ako kontrolu prepojenie požiadavky a odpovede. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).
Process-ID	Ne	String	Identifikátor obchodného alebo technického procesu, na základe ktorého možno párovať sadu dvojíc požiadaviek a odpovedí. Odporúča sa použiť formulár UUID (Universally Unique Ifentifier) verzie 4 (RFC4122).

2.9.2 Služba AISP (Dotazy k účtom, prehľad transakcií)

Kapitola definuje zoznam metód poskytovaných pre AISP.

2.9.2.1 Predpoklady pre používanie metód API pre službu AISP

a/ záznam TPP je na základe licenčného čísla (vrátane použitého prefixu) uvedeného v certifikáte, ktorý TPP používa pri komunikácii, nájdený v databáze IB

b/ nájdený záznam TPP je platný,

c/ TPP má vo svojom zázname povolenú službu AISP (táto informácia je súčasťou záznamu TPP v databáze IB, ktorý sa automaticky aktualizuje z NBS)

d/ registrovaná aplikácia TPP má povolenú službu AISP

e/ v certifikáte, ktorý používa TPP pri komunikácii je uvedená služba AISP

f/ TPP použil v hlavičke požiadavky access_token, na základe ktorého je na strane banky dohľadaný platný súhlas vytvorený disponentom.

g/ aplikácia TPP má v nájdenom súhlasu povolenú od disponenta službu AISP

2.9.2.2 Zoznam metód používaných pre službu AISP

Endpoint	Metoda	Popis
/api/v1/accounts/information	POST	prostredníctvom tejto služby dostane disponentom autorizovaná tretia strana prehľad zostatkov bankového účtu disponenta vedeného v danej banke
/api/v1/accounts/transactions	POST	prostredníctvom tejto služby dostane disponentom autorizovaná tretia strana prehľad transakcií
/api/v2/accounts	GET	služba na požiadanie vráti zoznam účtov, ktoré disponent uviedol v súhlasu k používaniu s konkrétnym TPP (nie zoznam všetkých účtov disponenta) bez zostatkov

2.9.2.3 Token pre AISP operácie

Pre AISP operácie bude používaný access_token získaný na základe autorizačného resource Authorization Code Grant s PKCE rozšírením popísaného v kapitole 2.8.1 alebo prípadne pozri [1], kapitola 4.1.

2.9.2.4 AISP operácia: Account Information

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api/**v1/accounts/information**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api/**v1/accounts/information**

Request

Metoda: accounts/information			
Názov atribútu	Formát	Povinný	Poznámka
iban	String (34)	Áno	IBAN účtu. Účet musí byť obsiahnutý v dohľadanom súhlase od disponenta

Response

Metoda: accounts/information				
Názov atribútu		Formát	Povinný	Poznámka
account	BaseCurrency	String (3)	Áno	Mena účtu (kód meny podľa ISO 4217 - 3 veľké písmená)
	Name	string	Áno	Názov účtu (meno klienta)
	ProductName	string	Nie	Názov produktu
	Type	enum	Nie	Skratka typu účtu podľa normy ISO 20022 - Cash Account Type Code – napr . • CACC - bežný účet
Balances		typ:ArrayOfAccountsInformationResponseBalance	Áno	Pole zostatkov

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.3.2.

2.9.2.4.1 Definícia typu ArrayOfAccountsInformationResponseBalance

Metóda: accounts/information - typ: ArrayOfAccountsInformationResponseBalance				
Názov atribútu		Formát	Povinný	Poznámka
amount	value	decimal (2 desatinné miesta)	Áno	Hodnota zostatku
	currency	String (3)	Áno	Kód meny zostatku podľa ISO 4217 - 3 veľké písmená

Metóda: accounts/information - typ: ArrayOfAccountsInformationResponseBalance				
creditDebitIndicator		enum	Áno	Skratka Indikátoru Kredit / Debet • CRDT (Kredit) • DBIT (Debet)
dateTime		dateTime	Áno	Dátum aktualizácie zostatku
typeCodeOrProprietary		enum	Áno	Typ zostatku • CLBD (aktuálny zostatok) • ITAV, ITBD (disponibilný zostatok)

2.9.2.5 AISP operácia: Account Transactions

Prostredníctvom tejto služby dostane disponentom autorizovaná tretia strana prehľad transakcií uskutočnených na bankovom účte zákazníka v rámci zadaného termínu. História transakcií zahŕňa iba transakcie, ktoré ovplyvňujú zostatok (rezervácie, zaúčtované transakcie). Transakcie sú zoradené od najnovšej po najstaršiu.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api/**v1/accounts/transactions**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api/**v1/accounts/transactions**

Request

Metóda: accounts/transactions			
Názov atribútu	Formát	Povinný	Poznámka
iban	String (34)	Áno	IBAN účtu. Účet musí byť obsiahnutý v dohľadanom súhlase od disponenta
dateFrom	dateTime	Nie	Dátum začiatku obdobia pre históriu transakcií. Predvolená hodnota je aktuálny deň
dateTo	dateTime	Nie	Dátum konca obdobia pre históriu transakcií. Predvolená hodnota je aktuálny deň
page	integer	Nie	Poradové číslo stránky vzhľadom na veľkosť stránky pre záznamovú sadu. Predvolená hodnota je 0 (prvá stránka).
pageSize	integer	Nie	Počet záznamov zahrnutých na jednej stránke pre zobrazenie. Predvolená hodnota je 50 záznamov. Maximálna povolená hodnota je 200 záznamov na stránku.
Status	Enum	Nie	Typ transakcie. Povolené typy: • BOOK (rezervácia) • INFO (zaúčtované transakcie) • ALL (všetky transakcie) Predvolená hodnota je ALL.

Response

Metóda: accounts/transactions			
Názov atribútu	Formát	Povinný	Poznámka
pageCount	integer	Nie	Celkový počet stránok
transactions	typ: ArrayOfAccountsTransactionsResponseTransaction	Áno	Pole transakcií

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.3.3.

2.9.2.5.1 Definícia typu ArrayOfAccountsTransactionResponseTransaction

Metóda: Account/Transaction - typ: ArrayOfAccountsTransactionsResponseTransaction				
Názov atribútu (každý stĺpec predstavuje jednu úroveň v JSON štruktúre)		Formát	Povinný	Poznámka
amount	value	decimal (2 desatinné miesta)	Áno	Hodnota čiastky transakcie
	currency	String (3)	Áno	Mena čiastky transakcie podľa ISO 4217 - 3 veľké písmená
creditDebitIndicator		enum	Áno	Skratka Indikátoru Kredit / Debet • CRDT (Kredit) • DBIT (Debet)
reversalIdentificator		boolean	Nie	Príznak určuje, či sa jedná o reverznú transakciu
status		enum	Áno	Typ transakcie. • BOOK (rezervácia) • INFO (zaúčtované transakcie)
bookingDate		dateTime	Povinné pre transakciu typu BOOK	Dátum rezervácie transakcie
valueDate		dateTime	Áno	Dátum valuty transakcie
bankTransactionCode		String	Nie	Kód kategórie typu transakcie zo zoznamu kódov SBA
transactionDetails		typ:AccountsTransactionsResponseTransactionDetail	Áno	Položky detailu transakcie

2.9.2.5.2 Definícia typu AccountsTransactionsResponseTransactionDetail

Metóda: Account/Transaction - typ: ArrayOfAccountsTransactionsResponseTransaction					
Názov atribútu (každý stĺpec predstavuje jednu úroveň v JSON štruktúre)			Formát	Povinný	Poznámka
additionalTransactionInformation			String	Nie	Popis bankovej transakcie
counterValueAmount	amount	value	Decimal (2 desatinné miesta)	Nie	Hodnota čiastky transakcie
		currency	String (3)	Nie	Mena čiastky transakcie podľa ISO 4217 - 3 veľké písmená
	currencyExchange	exchangeRate	Decimal (2 desatinné miesta)	Nie	Použitý výmenný kurz pre konverziu z inštruovanej meny na menu cieľového účtu
references	additionalTransactionInformation		String (35)	Nie	Jedinečné ID transakcie generované bankou
	chequeNumber		String (35)	Nie	Používané pri kartových transakciách Číslo karty vo formáte **** **** **** 1111
	endToEndIdentification		String (35)	Nie	Jedinečná identifikácia definovaná žiadateľom
	instructionIdentification		String (35)	Nie	Identifikácia platby generovaná klientom
	mandateIdentification		String (35)	Nie	Odkaz na mandát (referenčné číslo)
	transactionIdentification		String (35)	Nie	ID platby
relatedAgents	creditorAgent	financialInstitutionIdentification	String (11)	Nie	Identifikácia banky príjemcu, obvykle bankový identifikačný kód (BIC)
	deptorAgent	financialInstitutionIdentification	String (11)	Nie	Identifikácia banky platiteľa, obvykle bankový identifikačný kód (BIC)
relatedDates	acceptanceDateTime		Date	Nie	Dátum zadania transakcie (dátum prijatia transakcie v banke)
relatedParties	creditor	identification	String (35)	Nie	Identifikátor príjemcu (CID) v transakcii inkasa
		name	String	Nie	Meno príjemcu

Metóda: Account/Transaction - typ: ArrayOfAccountsTransactionsResponseTransaction					
	creditorAccount	identification	String (34)	Nie	Jedinečná identifikácia účtu príjemcu (IBAN)
	debtor	Name	String	Nie	Meno platiteľa
	debtorAccount	Identification	String (34)	Nie	Jedinečná identifikácia účtu platiteľa (IBAN)
	tradingParty	Identification	String (35)	Nie	Jedinečná identifikácia tretej strany. Pre kartové transakcie je tu uvádzané ID obchodníka
		merchantCode	String (4)	Nie	Kód kódu obchodníka (MCC) koordinovaný spoločnosťou MasterCard a Visa
		name	String	Nie	Meno tretej strany. Pre kartové transakcie je tu uvádzané obchodníka
remittanceInformation			String	Nie	Text pre príjemcu transakcie

2.9.2.6 AISP operácia: Account List

Testovacie prostredie

Endpoint: GET https://apitest.szrb.sk:98/api**/v2/accounts**

Produkčné prostredie

Endpoint: GET https://api.szrb.sk:98/api**/v2/accounts**

Request

Telo požiadavky neobsahuje žiadne atribúty.

Response

Metóda: accounts			
Názov atribútu	Formát	Povinný	Poznámka
creationDateTime	DateTime	Áno	Dátum a čas formátovaný podľa RFC3339, v ktorom bola konkrétna akcia vyžiadaná
accounts	typ: ArrayOfAccountInfo	Áno	Pole zostatkov

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.3.4.

2.9.2.6.1 Definícia typu ArrayOfAccountsInfo

Metóda: accounts - typ: ArrayOfAccountInfo				
Názov atribútu (každý stĺpec predstavuje jednu úroveň v JSON štruktúre)		Formát	Povinný	Poznámka
identification	iban	String (34)	Áno	IBAN účtu disponenta
name		String	Áno	Názov účtu (meno klienta)
productName		String	Nie	Názov produktu

Metóda: accounts - typ: ArrayOfAccountInfo				
type		Enum	Nie	Skratka typu účtu podľa normy ISO 20022 - Cash Account Type Code – napr . • CACC - bežný účet • LOAD – úverový účet • SVGS – sporiaci účet
baseCurrency		String (3)	Áno	Mena účtu (kód meny podľa ISO 4217 - 3 veľké písmená)
servicer	financialInstitutionIdentification	String (11)	Áno	BIC kód SZRB
consent		Array of string	Áno	Zoznam služieb, ktoré má TPP povolené disponentom v súhlase. Pole môže obsahovať nasledujúce reťazce: AISP, PISP, PIISP

2.9.3 Služby PISP (Vytvorenie platby, zisťovanie stavu platby, autorizácia platby, zrušenie platby)

Kapitola definuje zoznam metód poskytovaných pre PISP.

2.9.3.1 Predpoklady pre používanie metód API pre PISP

a/ záznam TPP je na základe licenčného čísla (vrátane použitého prefixu) uvedeného v certifikáte, ktorý TPP používa pri komunikácii, nájdený v databáze IB

b/ nájdený záznam TPP je platný,

c/ TPP má vo svojom zázname povolenú službu PISP (táto informácia je súčasťou záznamu TPP v databáze IB, ktorý sa automaticky aktualizuje z NBS)

d/ registrovaná aplikácia TPP má povolenú službu PISP

e/ v certifikáte, ktorý používa TPP pri komunikácii je uvedená služba PISP

f/ TPP použil v hlavičke požiadavky access_token, na základe ktorého je na strane banky dohľadaný platný súhlas vytvorený disponentom.

g/ aplikácia TPP má v nájdenom súhlasu povolenú od disponenta službu PISP

2.9.3.2 Zoznam metód používaných pre PISP

Endpoint	Metoda	Popis
/api/v1/payments/standard/iso	POST	**Standard payment initialization (XML)** - Prostredníctvom tejto metódy disponentom autorizovaná tretia strana iniciuje (vytvorí) SEPA príkaz z bankového účtu disponenta. Iniciovanie platby bude vykonané zaslaním súboru vo formáte XML (PAIN.001) v tele požiadavky.
/api/v1/payments/submission	POST	**Standard payment submission –** prostredníctvom tejto služby je umožnené TPP autorizovať platbu iniciovanú pomocou služby "Standard payment initialization"
/api/v1/payments/{orderId}/status	GET	**Payment order status –** prostredníctvom tejto služby je TPP umožnené zisťovanie stavu platobného príkazu
/api/v1/payments/{orderId}/rcp	DELETE	**Cancel payment -** prostredníctvom tejto služby je umožnené zrušenie platby, ktorá ešte nebola autorizovaná treťou stranou (tretia strana nepoužila metódu "Standard payment submission") a ktorá bola vytvorená prostredníctvom služby PISP Standard payment initialization (XML)
/api/v1/accounts/balanceCheck	POST	**Balance check** - prostredníctvom tejto metódy si TPP môže overiť, či má klient na bankovom účte, dostatok prostriedkov na zrealizovanie transakcie

2.9.3.3 PISP operácia: Standard payment initialization (XML)

Operácia umožňuje inicializáciu jednej SEPA platby vo formáte XML (PAIN.001.001.03). PISP odošle cez API požiadavku obsahujúcu jednu platbu založenú na štruktúre definovanej normou ISO20022 pain.001.001.03. Odoslaním tejto požiadavky sa na strane banky vytvorí SEPA platobný príkaz, ktorý sa vzťahuje k obchodnej transakcii medzi PSU a obchodníkom (TPP typu PISP).

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api**/v1/payments/standard/iso**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api**/v1/payments/standard/iso**

Request

Telo požiadavky obsahuje jednu SEPA platbu vo formáte xml: pain.001.001.03
Viz https://www.iso20022.org/documents/general/Payments_Maintenance_2009.zip

Response

Odpoveď, pokiaľ je príkaz uložený do databázy

Telo odpovede obsahuje opis zadanej platby vo formáte xml: pain.002.001.03

Metoda: payments/standard/iso				
Názov atribútu	Výskyt v XML struktuře odpovědi	Formát	Povinný	Poznámka
orderId	TxInfAndSts/AcctSvcrRef	String	Áno	Číslo príkazu vytvoreného v databáze Internet Bankingu
reasonCode	TxInfAndSts/StsRsnInf/Rsn	String	Nie	Status Reason Code podle ISO 20022 Viz: https://www.iso20022.org/sites/default/files/documents/External_code_lists/ExternalCodeSets_4Q2017_05Mar2018_v1.xls, (listy: 16-StatusReason, 60-ReceivedReason, 61-AcceptedReason, 62-PendingProcessingReason, 63-RejectedReason

Metoda: payments/standard/iso				
status	TxInfAndSts/TxSts	Enum	Áno	Status spracovania príkazu Status môže nadobúdať nasledujúce hodnoty: ACTC (vykonaná validácia položiek, príkaz čaká na autorizáciu klientom)
statusDateTime	GrpHdr/CreDtTm	dateTime	Nie	Dátum prijatia príkazu do banky

Chybové kódy		
HTTP Status	Error kód	Popis
400	**parameter_missing**	Chýba povinný parameter.
400	**parameter_invalid**	Nevalidná hodnota vstupného parametra.
500, 503	**server_error**	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.4.2.

2.9.3.4 PISP operácia: Standard payment submission

Operácia umožňuje tretej strane autorizáciu platby, ktorú táto TPP inicializovala.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/api**/v1/payments/submission**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api**/v1/payments/submission**

Request

Telo požiadavky neobsahuje žiadne atribúty.

Hlavička požiadavky musí obsahovať token "bearer token" (access_token), ktorý bude prepojený s práve autorizovaným príkazom s daným "orderId". Aby TPP získala tento access_token, musí predtým prebehnúť autorizácia danej platby disponentom (pozri kapitolu 2.9.3.4.2). Výsledkom tejto autorizácie je autorizačný kód (code), ktorý dostane TPP v odpovedi. Tento code následne TPP vymení pomocou Authorization code flow za daný access_token, previazaný s daným príkazom (pozri kapitolu 2.9.3.4.3).

Response (pokiaľ nedôjde pri spracovaní požiadavky k chybe)

Metoda: payments/submission			
Názov atributu	Formát	Povinné	Poznámka
orderId	String	Áno	Číslo príkazu vytvoreného v databáze Internet Bankingu
reasonCode	String	Nie	Status Reason Code podle ISO 20022 Viz: https://www.iso20022.org/sites/default/files/documents/External_code_lists/ExternalCodeSets_4Q2017_05Mar2018_v1.xls, (listy: 16-StatusReason, 60-ReceivedReason, 61-AcceptedReason, 62-PendingProcessingReason, 63-RejectedReason
status	Enum	Áno	Status příkazu Status môže nadobúdať nasledujúce hodnoty: › RJCT (Odmietnuté - Rejected) › PDNG (Autorizované - Authorized) › ACTC (K podpisu - WaitingForSignatures) › ACSP (Zpracováva sa - InProgress, Exportované - Exported) › ACSC (Akceptované bankovým systémom)
statusDateTime	dateTime	Nie	Datum přijetí příkazu do banky.

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.4.5.

2.9.3.4.1 Token pro PISP operáciu Autorizácia platby (Standard Payment submission)

Pre autorizáciu platby bude používaný access_token získaný na základe autorizačného resource Authorization Code Grant s PKCE rozšírením popísaného v kapitole 2.8.1 alebo prípadne pozri [1], kapitola 4.1.

Generovanie access_tokenu na základe Client Credentials Grant flow nie je v riešení podporované.

2.9.3.4.2 Autorizácia platby (disponentom - užívateľom aplikácie TPP)

Proces autorizácie platby disponentom musí PISP iniciovať po tom, čo sa po vytvorení (inicializácii) platby vráti v odpovedi z ASPSP (banky) číslo, pod akým sa daná platba na strane banky vytvorila (OrderId).

Testovacie prostredie

Endpoint: GET https://apitest.szrb.sk:98/auth/oauth/authorize

Produkčné prostredie

Endpoint: GET https://api.szrb.sk:98/auth/oauth**/authorize**

Request			
Atribut	Povinný	Typ	Popis
response_type	Áno	**code**	Povinný parameter. Hodnotou parametra je určené, aký typ autentizačného flow je požadovaný. V tomto prípade sa jedná o code grant. Pre autentizačný proces to znamená, že výsledkom tejto požiadavky bude jednorazový auth_code, ktorý TPP následne pomocou ďalšej požiadavky (metódou token) zamení za token access_token
client_id	Áno	String	Jedinečný identifikátor, ktorý banka vygenerovala pre aplikáciu TPP
redirect_uri	Áno	URL	URL kam je na konci presmerované flow autentizácie. Toto URL je stanovené už pri vydaní client_id a v rámci autentizácia je tento parameter validovaný proti URL zavedenému k client_id v zázname aplikácie registrované v banke. Hodnota by sa mala zhodovať s jednou z hodnôt uvedených v zázname registrované aplikácie.
Scope	Áno	String	Jedná sa o pole požadovaných scope (oprávnenia). V prípade PSD2 to môžu byť role AISP, PISP, PIISP. Napr. ak je TPP držiteľom viac oprávnenia, môže tu pre svoju aplikáciu požiadať len o jedno z nich alebo viac. Ak je použitých viac typov scope, sú oddelené medzerou.
state	Áno	Libovolný string [min 128 bits]	Parametrom sa zvyšuje bezpečnosť komunikácie pri presmerovaní. Chráni pred útokmi CSRF a odovzdáva informácie z aplikácie prostredníctvom toku autentizácie.
code_challenge	Áno	String	code_challenge = BASE64URL-ENCODE(SHA256(ASCII(code_verifier))) viz. zdroj [3] RFC 7636 (OAuth PKCE)
code_challenge_method	Áno	String	S256
request	Áno	JWT	Príklad použitia viz [8] kapitola 6.2.9

Súčasťou požiadavky o autorizáciu platby disponentom je **podpísaný JWT Request, ktorý obsahuje OrderId** (pre podpis vygenerovaného JWT musí TPP použiť client_secret, aby banka dokázala obsah JWT dešifrovať – client_secret je tajná informácia, ktorá je známa iba TPP a na bankovnej straně aplikácii IB, ktorá tento údaj pri registrácii TPP aplikácie vygenerovala).
Pri požiadavke o autorizácii platby disponentom bude disponent presmerovaný z aplikácie TPP na centrálnu autorizačnú stránku.

Potom, čo disponent vykoná dvojfázovú autorizáciu, zobrazí sa mu detail platby, ktorú musí autorizovať svojím autorizačným zariadením. Po autorizácii platby disponentom je v odpovedi vrátený autorizačný code, ktorý je previazaný s daným OrderId a platba čaká na autorizáciu treťou stranou.

Response			
Atribut	Povinný	Typ	Popis
Code	Áno	String	Jednorazový autorizačný kód
Id_token	Nie	JWT	**Nie je podporované**
State	Áno	String	Hodnota atribútu odovzdaného z TPP požiadavky

Chybové kódy

- Chybové kódy sú definovány podľa [1] RFC 6749, kapitola 4.1.2.1

Príklad použitia viz kapitola 3.2.4.3.

2.9.3.4.3 Získanie tokenu

Aby PISP mohol vykonať podpísanie vytvorené platby (/payments/submission), musí získať od banky access_token. Toto vykoná výmenou autorizačného Code, ktorý dostal v odpovedi požiadavky /Authorize, za daný access_token.

PISP zašle spoločne s týmto autorizačným kódom (ktorý musí byť uvedený v tele požiadavke) aj client_id a client_secret (ktoré však musí byť uvedené v hlavičke požiadavke zakódované pomocou Base64).

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/auth/oauth/**token**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/auth/oauth/**token**

Request			
Atribut	Povinný	Typ	Popis
code	Áno	string	Autorizačný code navrátený z autentizačného flow (code grant)
redirect_uri	Áno	URL	URL redirect zhodné s URL doručenom v autentizačnom requestu
grant_type	Áno	**authorization_code**	Podla existujúcej definície / zvyklosti OAuth2 bude táto hodnota authorization_code, ak dochádza k výmene code za access_token.
code_verifier	Áno	String	code_verifier slúži na generovanie code_challenge z predchádzajúcej žiadosti o minimálnej dĺžke 43 znakov a maximálnou dĺžkou 128 znakov

Response			
Atribut	Povinný	Typ	Popis
access_token	Áno	string	Krátkodobý token (platnosť tokenu je 3600s), ktorý slúži na autorizáciu requestu na API.
expires_in	Áno	number	Zostávajúci čas do expirácie access_tokenu - v sekundách.
token_type	Áno	String	Typ tokenu “Bearer”

Chybové kódy

- Chybové kódy sú definované podľa [1] RFC 6749, kapitola 5.2

Príklad použitia viz kapitola 3.2.4.4.

2.9.3.4.4 Autorizácia platby treťou stranou (TPP)

Potom, čo TPP získa access_token previazaný s daným príkazom, prevedie posledný krok - vytvorí požiadavku autorizácie danej platby (pozri kapitolu 2.9.3.4)

2.9.3.5 PISP operácia: Payment Order Status

Operácia poskytuje informácie o stave spracovania prijatej platobnej transakcie na základe parametra {orderId}.

Testovacie prostredie

Endpoint: GET https://apitest.szrb.sk:98/**v1/payments/{orderId}/status**

Produkčné prostredie

Endpoint: GET https://api.szrb.sk:98/api/**v1/payments/{orderId}/status**

Request

Telo požiadavky neobsahuje žiadne atribúty.

Response

Metóda: payments/{orderId}/status			
Názov atribútu	Formát	Povinný	Poznámka
orderId	String	Áno	Číslo príkazu vytvoreného v databáze Internet Bankingu
reasonCode	String	Nie	V položke sa uvádza informácia o skutočnom statuse, ktorý ma príkaz v Internet Bankingu. Jedná sa o dodatočnú informáciu k poľu „status“. Hodnota z tejto položky má význam predovšetkým v prípade, keď bude v položke „status“ uvedená hodnota „Others“ – (tzn. že pri spracovaní príkazu v Internet Bankingu je príkaz v stave, ktorý nie je pri spracovaní bežný (nie je obsiahnutý v množine statusov uvedených v poli „status“).
status	Enum	Áno	Status príkazu Status môže dosahovať nasledujúce hodnoty: • RJCT (Zrušené klientom / Odmietnuté bankou- Rejected) • PDNG (Autorizované - Authorized) • ACTC (K podpisu - WaitingForSignatures) • ACSP (Zpracováva sa - InProgress, Exportované - Exported) • ACSC (Akceptované bankovným systémom) • OTHR (rezerva)
statusDateTime	dateTime	Nie	Dátum prijatia príkazu do banky.

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.4.6.

2.9.3.6 PISP operácia: Cancel payment

Operácia umožňuje zrušiť platbu, ktorá bola iniciovaná prostredníctvom identického providera (tretej strany) typu PISP pomocou služby „Standard payment Initialization (XML)“. Platbu je možné zrušiť, kým TPP túto platbu neautorizuje službou „Payment Order Submission“.

Testovacie prostredie

Endpoint: DELETE https://apitest.szrb.sk:98/v1/payments/{orderId}/rcp

Produkčné prostredie

Endpoint: DELETE https://api.szrb.sk:98/api/v1/payments/{orderId}/rcp

Request

Telo požiadavky neobsahuje žiadne atribúty.

Response

Metóda: payments/{orderId}/status			
Názov atribútu	Formát	Povinný	Poznámka
orderId	String	Áno	Číslo príkazu zrušeného v databáze Internet Bankingu

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.4.7.

2.9.3.7 PISP operácia: Balance Check

Prostredníctvom tejto metódy si TPP môže overiť, či má klient na bankovom účte, dostatok prostriedkov na vykonanie platby.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98/**v1/accounts/balanceCheck**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api/**v1/accounts/balanceCheck**

Request

Metóda: accounts/balanceCheck					
Názov atribútu			Formát	Povinný	Poznámka
iban			String (34)	Áno	IBAN účtu. Účet musí byť obsiahnutý v dohľadanom súhlase od disponenta
creationDate			dateTime	Nie	Dátum a čas vytvorenia požiadavky podľa RFC 3339
amount	value		Decimal (2 desatinné miesta)	Nie	Hodnota čiastky transakcie
	currency		String (3)	Nie	Mena čiastky transakcie podľa ISO 4217 - 3 veľké písmená
instructionIdentification			string	Áno	Technická identifikácia platby generovaná na strane PISP
relatedParties	tradingParty	address	string	Nie	Adresa obchodníka (obvykle obsahuje zreťazenie názvu ulice, čísla ulice atď.).)
		countryCode	string	Nie	Dvojznakový kód zeme obchodníka podľa normy ISO3166
		identification	string	Nie	Jedinečná identifikácia tretej strany. Pre transakciu s kartou je tu uvedené číslo obchodníka.
		merchantCode	string	Nie	Kód kódu obchodníka (MCC) koordinovaný spoločnosťou MasterCard a Visa
		name	string	Nie	Meno tretej strany V prípade kartových transakcií sa tu uvádza meno obchodníka
references	chequeNumbe		string	Nie	V prípade kartových transakcií sa tu uvádza číslo karty vo formáte **** **** **** 1111
	holderName		string	Nie	Meno držiteľa karty

Response

Metóda: accounts/balanceCheck			
Názov atribútu	Formát	Povinný	Poznámka
response	Enum	Áno	Výsledok volania. Môže nadobúdať nasledujúce hodnoty: APPR (dostatočné finančné prostriedky na účte) DECL (nedostatočné prostriedky na účte)
dateTime	dateTime	Áno	Dátum a čas formátovaný podľa RFC3339, v ktorom bola akcia vyžiadaná

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.
500, 503	server_error	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.4.8.

2.9.4 Služba PIISP (Overenie dostatočných prostriedkov na účte)

Kapitola definuje zoznam metód poskytovaných pre PIISP.

2.9.4.1 Predpoklady pre používanie metód API pre PIISP

a/ záznam TPP je na základe licenčného čísla (vrátane použitého prefixu) uvedeného v certifikáte, ktorý TPP používa pri komunikácii, nájdený v databáze IB

b/ nájdený záznam TPP je platný,

c/ TPP má vo svojom zázname povolenú službu PIISP (táto informácia je súčasťou záznamu TPP v databáze IB, ktorý sa automaticky aktualizuje z NBS)

d/ registrovaná aplikácia TPP má povolenú službu PIISP

e/ v certifikáte, ktorý používa TPP pri komunikácii je uvedená služba PIISP

f/ TPP použil v hlavičke požiadavky access_token, na základe ktorého je na strane banky dohľadaný platný súhlas vytvorený disponentom.

g/ aplikácia TPP má v nájdenom súhlasu povolenú od disponenta službu PIISP

2.9.4.2 Zoznam metód používaných pre službu PIISP

Endpoint	Metoda	Popis
/api/v1/accounts/balanceCheck	POST	Balance check - prostredníctvom tejto metódy si TPP môže overiť, či má klient na bankovom účte, ku ktorému TPP vydala kartu, dostatok prostriedkov k zrealizovaniu transakcie kartou

2.9.4.3 Token pre PIISP operáciu

Pre PIISP operáciu bude používaný access_token získaný na základe autorizačného resource Authorization Code Grant s PKCE rozšírením popísaného v kapitole 2.8.1 alebo prípadne pozri [1], kapitola 4.1.

Generovanie access_tokenu na základe Client Credentials Grant flow, ktoré je v SBAS uvedené ako alternatívne rešenie (teda nepovinné), nie je v riešení podporované.

2.9.4.4 PIISP operácia: Balance Check

Prostredníctvom tejto metódy si TPP môže overiť, či má klient na bankovom účte, ku ktorému TPP vydala kartu, dostatok prostriedkov na zrealizovanie transakcie kartou.

Testovacie prostredie

Endpoint: POST https://apitest.szrb.sk:98**/v1/accounts/balanceCheck**

Produkčné prostredie

Endpoint: POST https://api.szrb.sk:98/api**/v1/accounts/balanceCheck**

Request

Metóda: Accounts/balanceCheck					
Názov atribútu			Formát	Povinný	Poznámka
iban			String (34)	Áno	IBAN účtu. Účet musí byť obsiahnutý v dohľadanom súhlase od disponenta
creationDate			dateTime	Nie	Dátum a čas vytvorenia požiadavky podľa RFC 3339
amount	value		Decimal (2 desatinné miesta)	Nie	Hodnota čiastky transakcie
	currency		String (3)	Nie	Mena čiastky transakcie podľa ISO 4217 - 3 veľké písmená
instructionIdentification			string	Áno	Technická identifikácia platby generovaná na strane PIISP
relatedParties	tradingParty	address	string	Nie	Adresa obchodníka (obvykle obsahuje zreťazenie názvu ulice, čísla ulice atď.).)
		countryCode	string	Nie	Dvojznakový kód zeme obchodníka podľa normy ISO3166

Metóda: Accounts/balanceCheck					
		identification	string	Nie	Jedinečná identifikácia tretej strany. Pre transakciu s kartou je tu uvedené číslo obchodníka.
		merchantCode	string	Nie	Kód kódu obchodníka (MCC) koordinovaný spoločnosťou MasterCard a Visa
		name	string	Nie	Meno tretej strany V prípade kartových transakcií sa tu uvádza meno obchodníka
references	chequeNumbe		string	Nie	V prípade kartových transakcií sa tu uvádza číslo karty vo formáte **** **** **** 1111
	holderName		string	Nie	Meno držiteľa karty

Response

Metóda: Account/balanceCheck			
Názov atribútu	Formát	Povinný	Poznámka
response	Enum	Áno	Výsledok volania. Môže nadobúdať nasledujúce hodnoty: APPR (dostatočné finančné prostriedky na účte) DECL (nedostatočné prostriedky na účte)
dateTime	dateTime	Áno	Dátum a čas formátovaný podľa RFC3339, v ktorom bola akcia vyžiadaná

Chybové kódy		
HTTP Status	Error kód	Popis
400	parameter_missing	Chýba povinný parameter.
400	parameter_invalid	Nevalidná hodnota vstupného parametra.

Chybové kódy		
500, 503	**server_error**	Chyba autorizačného servera.
Použitie ostatných http status kódov a chybových kódov podľa [1] RFC 6749, kapitola 5.2		

Príklad použitia viz kapitola 3.2.5.2.

Upozornenie: **grant_type "client_credentials"** uvedený v príklade v zdroji [8] kapitola 7.2.2 **nie je v tomto rešení podporovaný** – pozri kapitolu 2.9.4.3).

3. Prílohy

3.1 **JWT** token

JWT token je použitý v položke *request* pri odosielaní žiadosti „Autorizácia platby disponentom“ (kapitola 3.2.4.3). JWT token musí obsahovať identifikátor OrderId príkazu, ktorý je súčasťou odpovedi požiadavky **Standard payment initialization (XML)** (kapitola 3.2.4.2).

3.1.1 Príklad obsahu JWT tokenu

```
{
  "alg": "HS256",
  "typ": "JWT"
}
.
{
  "iss": "Test_3233376",
  "aud": "https://api.szrb.sk:98",
  "response_type": "code id_token",
  "client_id": "Test_3233376",
  "redirect_uri": "https://www.destination_redirect.uri",
  "scope": "PISP",
  "state": "VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ",
  "claims": {
    "id_token": {
      "orderId": {
        "value": "urn: Banka:order: 10004799",
        "essential": true
      }
    }
  }
}
.
{
<<Verify signature>>
}
```

3.1.2 Príklad vypočítaného tokenu JWT použitého do request

Obsah JWT tokenu uvedený v kapitole 3.1.1 a podpísaný pomocou client_secret s hodnotou „1234567890_AaBbCcDdEeFfGgHhIiJjKk“ vypadá v šifrovanom tvaru nasledovne:

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJUZXN0XzMyMzMzNzYiLCJhdWQiOiJodHRwczovL2FwaS5zenJiLnNrIiwicmVzcG9uc2VfdHlwZSI6ImNvZGUgaWRfdG9rZW4iLCJjbGllbnRfaWQiOiJUZXN0XzMyMzMzNzYiLCJyZWRpcmVjdF91cmkiOiJodHRwczovL3d3dy5kZXN0aW5hdGlvbl9yZWRpcmVjdC51cmkiLCJzY29wZSI6IlBJU1AiLCJzdGF0ZSI6IlZzSDBUaUFCMWQzdDd5UjZWdkQzMURwVVpFVnJCWEFRIiwiY2xhaW1zIjp7Imlkx3Rva2VuIjp7Im9yZGVySWQiOnsidmFsdWUiOiJ1cm46IEJhbmthOm9yZGVyOiAxMDAwNDc5OSIsImVzc2VudGlhbCI6dHJ1ZX19fX0.Gb18QDTIDxuYSm5KueardInLrjTKKwA6Rm7fHn46E_w
```

3.1.3 Popis parametrov použitých v JWT

JWT		
Časť	Parameter	Poznámka

Header	"alg"	typ použitej šifrovacej metódy V rámci riešenia je podporovaný iba algoritmus ***"HS256"***
	"typ"	typ tokenu V rámci riešenia je podporovaný iba typ ***"JWT"***
Payload	"iss"	tu musí byť uvedená hodnota client_id z posielanej požiadavky na autorizáciu platby disponentom (identifikátor klientskej aplikácie TPP)
	"aud"	tu musí byť uvedené nasledujúce url: **"**https://api.szrb.sk:98**"**
	"response_type"	tu musí byť uvedený reťazec uvedený v špecifikácii štandardu: **"code id_token"**
	"client_id"	tu musí byť uvedená hodnota client_id z posielanej požiadavky na autorizáciu platby disponentom (identifikátor klientskej aplikácie TPP). Hodnota tejto položky je rovnaká ako v parametru "iss"
	"redirect_uri"	Tu musí byť uvedené redirect_url z posielanej požiadavky, na ktorú bude klient presmerovaný späť po autorizácii platby (táto URL adresa musí byť rovnaká s URL, ktorá je registrovaná pre aplikáciu TPP z procesu Enroll (pozri kapitolu 2.7))
	"scope"	Tu musí byť uvedená hodnota parametra „scope“ z posielanej požiadavky.
	"state"	Tu musí byť uvedená hodnota parametra „state“ z posielanej požiadavky.
	"claims"	Štruktúrovaný parameter, ktorý obsahuje OrderId. V parametri „claims“ je možné v parametri **"value"** použiť *Banka:* (rovnako ako v príkladu uvedeného v SBA) alebo *Privatbanka:* Za parametrom **order:** musí byť uvedený identifikátor autorizovaného príkazu, ktorý je súčasťou odpovedi požiadavky na inicializáciu platby (pozri kapitolu **Chyba! Nenašiel sa žiaden zdroj odkazov.**) v atribúte: `< GrpHdr><MsgId>???</MsgId>` **Ukážka štruktúrovaného parametru claims:** `"claims": { "id_token": { "orderId": { "value": "urn: Banka:order: 10004796", "essential": true } } }`
Verify signature		V tejto časti sa používa podpis generovaný metódou HMACSHA256. Ako šifrovací kľúč sa používa client_secret, ktorý TPP prijala v odpovedi pri registrácii svojej aplikácie (pozri kapitolu **Chyba! Nenašiel sa žiaden z droj odkazov.**). Príklad funkcie použitej pre výpočet Verify signature, ktorý je potom pripojený ako posledná časť do JWT (za oddeľovač - bodku). HMACSHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload), client_secret)

3.2 Príklady http (request / response)

3.2.1 (api/enroll) Registračné resource

3.2.1.1 (POST api/enroll) Automatické generovanie technických identifikátorov

HTTP request:

Header

```
POST api/enroll HTTP/1.1
Host: api.szrb.sk:98
Content-Type: application/json;charset=UTF-8
```

Body

```
{
   "redirect_uris":
      ["http://www.tpp.sk/TPPTest/Home/LoginReturn",
      "http://www.tpp.sk/TPPTest/Home/AuthorizationReturn"],
   "client_name": "Moja_app",
   "client_type": "confidential",
   "logo_uri": "https://www.tpp.sk/logo.png",
   "contacts": ["info@tpp.sk"],
   "scopes": ["AISP", "PISP"],
   "licence_number": "11223344"
}
```

HTTP response:

Header

```
HTTP/1.1 201 Created
Content-Type: application/json;charset=UTF-8
Cache-Control: no-store
Pragma: no-cache
```

Body

```
{
    "client_id": "Moja_app_3263351",
    "client_secret":
"Q6GOgsq6Kc7EXzyoxObzlgVxKOGGJifSWpUL6BOJNIni2qHHyrtgm9H6xlDAxYOCkdupBkbEqbFtKytwxCgGkg",
    "client_secret_expires_at": "0",
    "api_key": "NOT PROVIDED",
    "redirect_uris": [
      "http://www.tpp.sk/TPPTest/Home/LoginReturn",
      "http://www.tpp.sk/TPPTest/Home/AuthorizationReturn"
    ],
    "client_name": "Moja_app",
    "client_name#en-US": null,
    "client_type": "confidential",
    "logo_uri": "https://www.tpp.sk/logo.png",
    "contacts": [
        "info@tpp.sk"
    ],
    "scopes": [
        "AISP",
        "PISP"
    ],
    "licence_number": "11223344"
}
```

3.2.1.2 (PUT api/enroll/{client_id}) Zmena registračných údajov

HTTP request:

Header

```
PUT api/enroll/Moja_app_3263351 HTTP/1.1
Host: api.szrb.sk:98
Content-Type: application/json;charset=UTF-8
```

Body

```
{
   "redirect_uris":
      ["http://www.tpp.sk/TPPTest/Home/LoginReturn",
      "http://www.tpp.sk/TPPTest/Home/AuthorizationReturn"],
   "client_name": "Moja_app",
   "client_type": "confidential",
   "contacts": ["info@tpp.sk", "contact@tpp.sk"],
   "scopes": ["AISP", "PISP"]
}
```

HTTP response:

Header

```
HTTP/1.1 200
Content-Type: application/json;charset=UTF-8
Cache-Control: no-cache
Pragma: no-cache
```

Body

```
{
    "client_id": "Moja_app_3263351",
    "client_secret_expires_at": "0",
    "api_key": "NOT_PROVIDED",
    "redirect_uris": [
      "http://www.tpp.sk/TPPTest/Home/LoginReturn",
      "http://www.tpp.sk/TPPTest/Home/AuthorizationReturn"
    ],
    "client_name": "Moja_app",
    "client_name#en-US": null,
    "logo_uri": null,
    "contacts": [
        "info@tpp.sk",
        "contact@tpp.sk"
    ],
    "scopes": [
        "AISP",
        "PISP"
    ],
    "client_type": "confidential"
}
```

3.2.1.3 (DELETE api/enroll/{client_id}) Zmazanie aplikácie

HTTP request:

Header

```
DELETE api/enroll/Moja_app_3263351 HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
```

HTTP response:

```
HTTP/1.1 204 No content
```

3.2.1.4 (POST api/enroll/{client_id}/renewSecret) Žiadosť o nový client_secret

HTTP request:

Header

```
POST api/enroll/Moja_app_3263351/renewSecret HTTP/1.1
Host: api.szrb.sk:98
Content-Type: application/json;charset=UTF-8
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "client_id": "Moja app 3263351",
    "client_secret": "-
29Jcq7LgvpgPB2Hqy6PvxE7hKxUbqy2EwhzAgLujG6faqqo6mgPMmKji_vBwqQgOTUCfIUdtDQ",
    "client_secret_expires_at": "0"
}
```

3.2.2 (auth/oauth) Autentizácia a Autorizácia requestu

3.2.2.1 (GET auth/oauth/authorize) Autorizačný resource

HTTP request:

Header

```
GET auth/oauth/authorize HTTP/1.1
Host: api.szrb.sk:98
Content-Type: application/x-www-form-urlencoded
response_type=code&
scope=AISP&
client_id=Moja_app_3263351&
state=VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ&
redirect_uri=http://www.tpp.sk/TPPTest/Home/LoginReturn&
login_hint=&
code_challenge=oO77bZ2WVsphzUSIihF1VUB2H0AE5auo8uP_x8axjW0&
code_challenge_method=S256
```

HTTP response:

Header

```
HTTP/1.1 303 See Other
Content-Type: application/x-www-form-urlencoded
Location:
http://www.tpp.sk/TPPTest/Home/LoginReturn?code=gCyAymoimg0L1bEI&state=VsH0TiAB1d3t7yR6VvD31Dp
UZEVrBXAQ
```

Klient (PSU) je po autentifikácii presmerovaný späť na poskytovateľa (v našom prípade AISP). Súčasťou URL je parameter *state* a autorizačný kód, ktorý potom tretia strana používa na výmenu za dvojicu tokenov (access_token a refresh_token).

3.2.2.2 (POST auth/oauth/token) Získanie tokenov

HTTP request:

Header

```
POST auth/oauth/token HTTP/1.1
Host: api.srzb.sk
Content-Type: application/x-www-form-urlencoded
Authorization: Basic
TmV3X2NlcnRfMzIzMzM3Njo1VzVWTFF5ajNGWGY5cVBUZHo3cTJ5UE4wZ1dVWkFmeDNmZnNpQXBGX3Z0MDE2MzNncEd4cU
9zTGdoUnRldUtiWXN6VW5Ea3FXQlpFc0VocXpQMlJYQQ //Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=authorization_code&
code=gCyAymoimg0L1bEI&
redirect_uri=http://www.tpp.sk/TPPTest/Home/LoginReturn&
code_verifier=yDWNhLugLI3BqUvXDYWE3DPrggSEyXCR
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "access_token":
"RWZzdEJDdEZXZXlEU0puOWlKWUswSEF4RWNvdUxGY1AycnRxeHdPRjhPa3dsbHFETllOeHdV",
    "expires_in": 3600,
    "token_type": "Bearer",
    "refresh_token":
"V2FHUFkxYjhnSUhvaE5wcjZ1QmlnQzFtZDdjSmxoYnlHRmpRb2xUQUhXVFZ1QlVHcFg0YkM2",
    "scope": "AISP PISP"
}
```

3.2.2.3 (POST auth/oauth/token) Obnovenie Access tokenu

HTTP request:

Header

```
POST auth/oauth/token HTTP/1.1
Host: api.srzb.sk
Content-Type: application/x-www-form-urlencoded
Authorization: Basic
TmV3X2NlcnRfMzIzMzM3Njo1VzVWTFF5ajNGWGY5cVBUZHo3cTJ5UE4wZ1dVWkFmeDNmZnNpQXBGX3Z0MDE2MzNncEd4cU
9zTGdoUnRldUtiWXN6VW5Ea3FXQlpFc0VocXpQMlJYQQ //Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=refresh_token&
refresh_token=V2FHUFkxYjhnSUhvaE5wcjZ1QmlnQzFtZDdjSmxoYnlHRmpRb2xUQUhXVFZ1QlVHcFg0YkM2&
scope=AISP PISP
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "access_token":
"WlRJNVBaUDlHanl5QVpncmJqYjlJcU1HTFBnallYd1cyZU5NMGZxZEhJa1RYeHAxdkh6Ymd0",
    "expires_in": 3600,
    "token_type": "Bearer",
    "refresh_token":
"V2FHUFkxYjhnSUhvaE5wcjZ1QmlnQzFtZDdjSmxoYnlHRmpRb2xUQUhXVFZ1QlVHcFg0YkM2",
    "scope": "AISP PISP"
}
```

3.2.3 (api/) AISP

3.2.3.1 Všeobecná definícia hlavičiek

HTTP request:

Header

```
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

3.2.3.2 (POST api/v1/accounts/information) Account Information

HTTP request:

Header

```
POST api/v1/accounts/information HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

Body

```
{
  "iban": "SK8230000000000123123123"
}
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body

```
{
    "account": {
        "name": "Acc 00002456",
        "productName": "Vkladový/Bežný účet",
        "type": "CACC",
        "baseCurrency": "EUR",
        "openDate": "2009-03-16T00:00:00"
    },
    "balances": [
        {
            "typeCodeOrProprietary": "ITAV",
            "amount": {
                "value": 10000000.00,
                "currency": "EUR"
            },
            "creditDebitIndicator": "CRDT",
            "dateTime": "2014-09-23T00:00:00"
        },
        {
            "typeCodeOrProprietary": "ITBD",
            "amount": {
                "value": 10000000.00,
                "currency": "EUR"
            },
            "creditDebitIndicator": "CRDT",
            "dateTime": "2014-09-23T00:00:00"
        }
    ]
}
```

3.2.3.3 (POST api/v1/accounts/transactions) Account transactions

HTTP request:

Header

```
POST api/v1/accounts/transactions HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX
Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

Body

```
{
  "iban": "SK8230000000000123123123",
  "dateFrom": "2021-11-01",
  "dateTo": "2021-11-30",
  "pageSize": 50,
  "page": 0
}
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body

```
{
    "pageCount": 1,
    "transactions": [
        {
            "amount": {
                "value": 75.74,
                "currency": "EUR"
            },
            "creditDebitIndicator": "CRDT",
            "reversalIdentificator": false,
            "status": "INFO",
            "bookingDate": "2021-11-18T00:00:00",
            "valueDate": "2021-11-18T00:00:00",
            "paymentDate": "2021-11-18T00:00:00",
            "bankTransactionCode": null,
            "transactionDetails": {
                "references": {
                    "additionalTransactionInformation": "",
                    "instructionIdentification": null,
                    "endToEndIdentification": "/VS/SS/KS",
                    "transactionIdentification": null,
                    "mandateIdentification": null,
                    "chequeNumber": null
                },
                "counterValueAmount": null,
                "relatedParties": {
                    "debtor": {
                        "name": "Vnutrobankovy ucet"
                    },
                    "debtorAccount": {
                        "identification": "SK2081200000000002237060"
                    },
                    "creditor": {
                        "name": "Account 00002456",
                        "identification": null
                    },
                    "creditorAccount": {
                        "identification": "SK8230000000000123123123"
                    },
                    "tradingParty": null
                },
                "relatedAgents": {
                    "deptorAgent": {
                        "financialInstitutionIdentification": "BSLOSK22XXX"
```

```
                    },
                    "creditorAgent": {
                        "financialInstitutionIdentification": "SLZBSKBAXXX"
                    }
                },
                "remittanceInformation": "Informácia o platbe",
                "relatedDates": null,
                "additionalTransactionInformation": "Ďalšie informácie o platbe"
            }
        }
    ]
}
```

3.2.3.4 (GET api/v2/accounts) List of accounts

HTTP request:

Header

```
GET api/v2/accounts HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10 14 3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body


```
{
    "creationDateTime": "2022-06-06T13:31:23.5937196+02:00",
    "accounts": [
        {
            "identification": {
                "iban": "SK8230000000000123123123"
            },
            "name": "Acc 00002456",
            "productName": "Bežný účet",
            "type": "CACC",
            "baseCurrency": "EUR",
            "servicer": {
                "financialInstitutionIdentification": "SLZBSKBAXXX"
            },
            "consent": [
                "AISP",
                "PISP"
            ]
        }
    ]
}
```

3.2.4 (api/) PISP

3.2.4.1 Všeobecná definícia hlavičiek

HTTP request:

Header

```
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

3.2.4.2 (POST api/v1/payments/standard/iso) Standard payment initialization (XML)

HTTP request:

Header

```
POST api/v1/payments/standard/iso HTTP/1.1
Host: api.srzb.sk
Content-Type: application/xml;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10 14 3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

Body

```
<?xml version="1.0" encoding="utf-8"?>
<Document xmlns="urn:iso:std:iso:20022:tech:xsd:pain.001.001.03"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <CstmrCdtTrfInitn>
    <GrpHdr>
      <MsgId>1000027313</MsgId>
      <CreDtTm>2022-05-31T07:43:26</CreDtTm>
      <NbOfTxs>1</NbOfTxs>
      <CtrlSum>23.00</CtrlSum>
      <InitgPty>
        <Nm>Meno</Nm>
        <Id>
          <OrgId>
            <Othr>
              <Id>ABC83902001</Id>
            </Othr>
          </OrgId>
        </Id>
      </InitgPty>
    </GrpHdr>
    <PmtInf>
      <PmtInfId>1000027313</PmtInfId>
      <PmtMtd>TRF</PmtMtd>
      <PmtTpInf>
        <SvcLvl>
          <Cd>URNS</Cd>
        </SvcLvl>
      </PmtTpInf>
      <ReqdExctnDt>2022-06-06</ReqdExctnDt>
      <Dbtr>
        <Nm>meno</Nm>
        <PstlAdr>
          <Ctry>SK</Ctry>
          <AdrLine>Ulica 5</AdrLine>
        </PstlAdr>
      </Dbtr>
      <DbtrAcct>
        <Id>
          <IBAN>SK8230000000000123123123</IBAN>
        </Id>
        <Ccy>EUR</Ccy>
      </DbtrAcct>
      <DbtrAgt>
        <FinInstnId>
          <BIC>SLZBSKBAXXX</BIC>
          <PstlAdr>
            <Ctry>SK</Ctry>
          </PstlAdr>
        </FinInstnId>
      </DbtrAgt>
      <ChrgBr>SHAR</ChrgBr>
      <CdtTrfTxInf>
        <PmtId>
          <InstrId>02-5001F1100000012015</InstrId>
          <EndToEndId>/VS1111111111/KS0968/SS2222222222</EndToEndId>
        </PmtId>
        <PmtTpInf>
          <InstrPrty>NORM</InstrPrty>
        </PmtTpInf>
        <Amt>
```

```
          <InstdAmt Ccy="EUR">23.00</InstdAmt>
        </Amt>
        <CdtrAgt>
          <FinInstnId>
            <BIC>BSLOSK22</BIC>
            <PstlAdr>
              <Ctry>SK</Ctry>
            </PstlAdr>
          </FinInstnId>
        </CdtrAgt>
        <Cdtr>
          <Nm>Meno príjemcu</Nm>
          <PstlAdr>
            <Ctry>SK</Ctry>
            <AdrLine>Ulica 45</AdrLine>
            <AdrLine>Bratislava Slovenská republika</AdrLine>
          </PstlAdr>
        </Cdtr>
        <CdtrAcct>
          <Id>
            <IBAN>SK0281201020073084015915</IBAN>
          </Id>
        </CdtrAcct>
        <RmtInf>
          <Ustrd>/INV/1234567890</Ustrd>
        </RmtInf>
      </CdtTrfTxInf>
    </PmtInf>
  </CstmrCdtTrfInitn>
</Document>
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/xml;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body

```
<Document xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="urn:iso:std:iso:20022:tech:xsd:pain.002.001.03">
    <CstmrPmtStsRpt>
        <GrpHdr>
            <MsgId>10004851</MsgId>
            <CreDtTm>2022-06-06T13:58:25.3382882+02:00</CreDtTm>
            <DbtrAgt>
                <FinInstnId>
                    <BIC>SLZBSKBAXXX</BIC>
                </FinInstnId>
            </DbtrAgt>
        </GrpHdr>
        <OrgnlGrpInfAndSts>
            <OrgnlMsgId>1000027313</OrgnlMsgId>
            <OrgnlCreDtTm>2022-05-31T07:43:26</OrgnlCreDtTm>
            <OrgnlNbOfTxs>1</OrgnlNbOfTxs>
            <OrgnlCtrlSum>23.00</OrgnlCtrlSum>
            <NbOfTxsPerSts>
                <DtldNbOfTxs>1</DtldNbOfTxs>
                <DtldSts>ACTC</DtldSts>
            </NbOfTxsPerSts>
        </OrgnlGrpInfAndSts>
        <OrgnlPmtInfAndSts>
            <OrgnlPmtInfId>1000027313</OrgnlPmtInfId>
            <OrgnlNbOfTxs>1</OrgnlNbOfTxs>
            <OrgnlCtrlSum>23.00</OrgnlCtrlSum>
            <TxInfAndSts>
                <StsId>10004851</StsId>
                <OrgnlInstrId>02-5001F1100000012015</OrgnlInstrId>
                <OrgnlEndToEndId>/VS1111111111/KS0968/SS2222222222</OrgnlEndToEndId>
                <StsRsnInf>
                    <Orgtr>
                        <Id>
                            <OrgId>
```

```
                            <BICOrBEI>SLZBSKBAXXX</BICOrBEI>
                        </OrgId>
                    </Id>
                </Orgtr>
            </StsRsnInf>
            <AcctSvcrRef>10004851</AcctSvcrRef>
            <OrgnlTxRef>
                <Amt>
                    <InstdAmt Ccy="EUR">23.00</InstdAmt>
                </Amt>
                <ReqdExctnDt>2022-06-06</ReqdExctnDt>
                <RmtInf>
                    <Ustrd>/INV/1234567890</Ustrd>
                </RmtInf>
                <Dbtr>
                    <Nm>Meno</Nm>
                </Dbtr>
                <DbtrAcct>
                    <Id>
                        <IBAN>SK8230000000000123123123</IBAN>
                    </Id>
                </DbtrAcct>
                <DbtrAgt>
                    <FinInstnId>
                        <BIC>SLZBSKBAXXX</BIC>
                    </FinInstnId>
                </DbtrAgt>
                <CdtrAgt>
                    <FinInstnId>
                        <BIC>BSLOSK22</BIC>
                    </FinInstnId>
                </CdtrAgt>
                <Cdtr>
                    <Nm>Meno príjemcu</Nm>
                </Cdtr>
                <CdtrAcct>
                    <Id>
                        <IBAN>SK0281201020073084015915</IBAN>
                    </Id>
                </CdtrAcct>
            </OrgnlTxRef>
        </TxInfAndSts>
    </OrgnlPmtInfAndSts>
</CstmrPmtStsRpt>
</Document>
```

3.2.4.3 (GET auth/oauth/authorize) Autorizácia platby disponentom

HTTP request:

Header

```
GET auth/oauth/authorize HTTP/1.1
Host: api.srzb.sk
Content-Type: application/x-www-form-urlencoded

response_type=code&
client_id=Moja_app_3263351&
scope=PISP&
state=VsH0TiAB1d3t7yR6VvD31DpUZEVrBXAQ&
redirect_uri=http://www.tpp.sk/TPPTest/Home/LoginReturn&
code_challenge=oO77bZ2WVsphzUSIihF1VUB2H0AE5auo8uP_x8axjW0&
code_challenge_method=S256&
request=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJUZXN0XzMyMzMzNzYiLCJhdWQiOiJodHRwczovL
2FwaS5zenJiLnNrIiwicmVzcG9uc2VfdHlwZSI6ImNvZGUgaWRfdG9rZW4iLCJjbGllbnRfaWQiOiJUZXN0XzMyMzMzNzY
iLCJyZWRpcmVjdF91cmkiOiJodHRwczovL3d3dy5kZXN0aW5hdGlvbl9yZWRpcmVjdC51cmkiLCJzY29wZSI6IlBJU1AiL
CJzdGF0ZSI6IlZzSDBUaUFCMWQzdDd5UjZWdkQzMURwVVpFVnJCWEFRIiwiY2xhaW1zIjp7ImlkX3Rva2VuIjp7Im9yZGV
ySWQiOnsidmFsdWUiOiJ1cm46IEJhbmthOm9yZGVyOiAxMDAwNDc5OSIsImVzc2VudGlhbCI6dHJ1ZX19fX0.Gb18QDTID
xuYSm5KueardInLrjTKKwA6Rm7fHn46E_w
```

Súčasťou požiadavky o autorizáciu platby disponentom je podpísaný JWT Request, ktorý obsahuje OrderId (viď. kapitolu 3.1). Pri požiadavke o autorizáciu platby disponentom bude disponent presmerovaný z aplikácie TPP na autorizačnú stránku banky.

HTTP response:

Header

```
HTTP/1.1 303 See Other
Content-Type: application/x-www-form-urlencoded
Location:
http://www.tpp.sk/TPPTest/Home/LoginReturn?code=IDPCyAymoimg0L1bEI&state=VsH0TiAB1d3t7yR6VvD31
DpUZEVrBXAQ
```

Potom, čo disponent vykoná dvojfaktorovú autentifikáciu, zobrazí sa mu detail platby, ktorú musí autorizovať (rovnako ako v Internetbankingu) svojím autorizačným zariadením. Po autorizácii platby disponentom je v odpovedi vrátený parameter *state* a v parametri *code* autorizačný code, ktorý je previazaný s daným OrderId a platba čaká na autorizáciu treťou stranou.

3.2.4.4 (POST auth/oauth/token) Získanie tokenu pre požiadavku Standard payment submission

HTTP request:

Header

```
POST auth/oauth/token HTTP/1.1
Host: api.srzb.sk
Content-Type: application/x-www-form-urlencoded
Authorization: Basic
TmV3X2NlcnRfMzIzMzM3Njo1VzVWTFF5ajNGWGY5cVBUZHo3cTJ5UE4wZ1dVWkFmeDNmZnNpQXBGX3Z0MDE2MzNncEd4cU
9zTGdoUnRldUtiWXN6VW5Ea3FXQlpFc0VocXpQMlJYQQ //Basic BASE64(CLIENT_ID + ":" + CLIENT_SECRET)
```

Body

```
grant_type=authorization_code&
code=IDPCyAymoimg0L1bEI&
redirect_uri=http://www.tpp.sk/TPPTest/Home/LoginReturn&
code_verifier=yDWNhLugLI3BqUvXDYWE3DPrggSEyXCR
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "access_token": "WmszcjFUOWJGRWpkRVlhSjhFZTRPekZHaGJGMU1GdcEFtWU5PdkI0aU91N3l5",
    "expires_in": 3600,
    "token_type": "Bearer",
    "refresh_token": null,
    "scope": "PISP"
}
```

3.2.4.5 (POST api/v1/payments/paymentSubmission) Standard payment submission

HTTP request:

Header

```
POST api/v1/payments/paymentSubmission HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer WmszcjFUOWJGRWpkRVlhSjhFZTRPekZHaGJGMU1GdcEFtWU5PdkI0aU91N3l5

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "orderId": "10004851",
    "status": "PDNG",
    "reasonCode": "Authorized",
    "statusDateTime": "2022-06-06T14:40:08.2478907+02:00"
}
```

3.2.4.6 (GET api/v1/payments/{orderId}/status) Payment order status

HTTP request:

Header

```
GET api/v1/payments/10004851/status HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body

```
{
    "orderId": "10004851",
    "status": "ACSP",
    "reasonCode": "Processing",
    "statusDateTime": "2022-06-06T13:58:25.3382882"
}
```

3.2.4.7 (DELETE api/v1/payments/{orderId}/rcp) Cancel payment

HTTP request:

Header

```
DELETE api/v1/payments/10004851/rcp HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

Body

```
{
    "orderId": "10004851"
}
```

3.2.4.8 (POST api/v1/accounts/balanceCheck) Balance check

HTTP request:

Header

```
POST api/v1/accounts/balanceCheck HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

Body

```
{
  "instructionIdentification": "9b76608457de48b2be531bd2804ae0b7",
  "creationDateTime": "2019-05-27T18:00:00+01:00",
  "iban": "SK8230000000000123123123",
  "amount": {
    "value": 123.56,
    "currency": "EUR"
  },
  "relatedParties": {
    "tradingParty": {
      "identification": "AAA-GG-SSSS",
      "name": "Merchant ID",
      "address": "Ulica 123123, Mesto",
      "countryCode": "SK",
      "merchantCode": "3370"
    }
  },
  "references": {
    "chequeNumber": "**** **** **** 1111",
    "holderName": "Jan Dopek"
  }
}
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "response": "APPR",
    "dateTime": "2022-06-06T15:14:50.0564144+02:00"
}
```

3.2.5 (api/) PIISP

3.2.5.1 Všeobecná definícia hlavičiek

HTTP request:

Header

```
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

HTTP response:

Header

```
Content-Type: application/json;charset=UTF-8
Response-ID: 7deb90a9-9900-4c90-a91c-3ecc888c2c88
```

3.2.5.2 (POST api/v1/accounts/balanceCheck) Balance check

HTTP request:

Header

```
POST api/v1/accounts/balanceCheck HTTP/1.1
Host: api.srzb.sk
Content-Type: application/json;charset=UTF-8
Authorization: Bearer IDWJJBCHQ5DZJWEMO7ZWM4DLYWOFWKXX

Request-ID: c2c48fc8-1f79-4934-a47b-56d61a28f351
PSU-IP-Address: 192.168.0.100
PSU-Device-OS: iOS 12.1.4
PSU-User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/72.0.3626.109 Safari/537.36
```

Body

```
{
  "instructionIdentification": "9b76608457de48b2be531bd2804ae0b7",
  "creationDateTime": "2019-05-27T18:00:00+01:00",
  "iban": "SK8230000000000123123123",
  "amount": {
    "value": 123.56,
    "currency": "EUR"
  },
  "relatedParties": {
    "tradingParty": {
      "identification": "AAA-GG-SSSS",
      "name": "Merchant ID",
      "address": "Ulica 123123, Mesto",
      "countryCode": "SK",
      "merchantCode": "3370"
    }
  },
  "references": {
    "chequeNumber": "**** **** **** 1111",
    "holderName": "Jan Dopek"
  }
}
```

HTTP response:

Header

```
HTTP/1.1 200 ok
Content-Type: application/json;charset=UTF-8
```

Body

```
{
    "response": "APPR",
    "dateTime": "2022-06-06T15:14:50.0564144+02:00"
}
```

4. Zdroje

1. *RFC 6749 - The OAuth 2.0 Authorization Framework*, [online]. The Internet Engineering Task Force, October 2012. WWW: https://tools.ietf.org/html/rfc6749
2. *RFC 6750 - The OAuth 2.0 Authorization Framework: Bearer Token Usage,* [online]. The Internet Engineering Task Force, October 2012. WWW: https://tools.ietf.org/html/rfc6750
3. *RFC 7636 - Proof Key for Code Exchange by OAuth Public Clients,* [online]. The Internet Engineering Task Force, September 2015. WWW: https://tools.ietf.org/html/rfc7636
4. *RFC 7519 - JSON Web Token (JWT)*, [online]. The Internet Engineering Task Force, May 2015. WWW: https://tools.ietf.org/html/rfc7519
5. *RFC 7515 - JSON Web Signature (JWS)*, [online]. The Internet Engineering Task Force, May 2015. WWW: https://tools.ietf.org/html/rfc7515
6. *Slovak Banking API Standard, SBA et al.,* [online]. WWW: http://docs.sbaonline.apiary.io/#
7. *ISO 20022 Financial Services - Universal financial industry message scheme,* [online]. International Organization for Standardization. WWW: https://www.iso20022.org/
8. *Slovak Banking API Standard*, dokument. WWW: https://www.sbaonline.sk/wp-content/uploads/2020/03/slovak-banking-api-standard-2_0.pdf