

## SANDBOX ACCOUNT DIRECT ACCESS API

Account Direct Access API (ADAA) service provides a SANDBOX version as a demonstration for development and testing purposes. This document describes how to use ADAA SANDBOX.

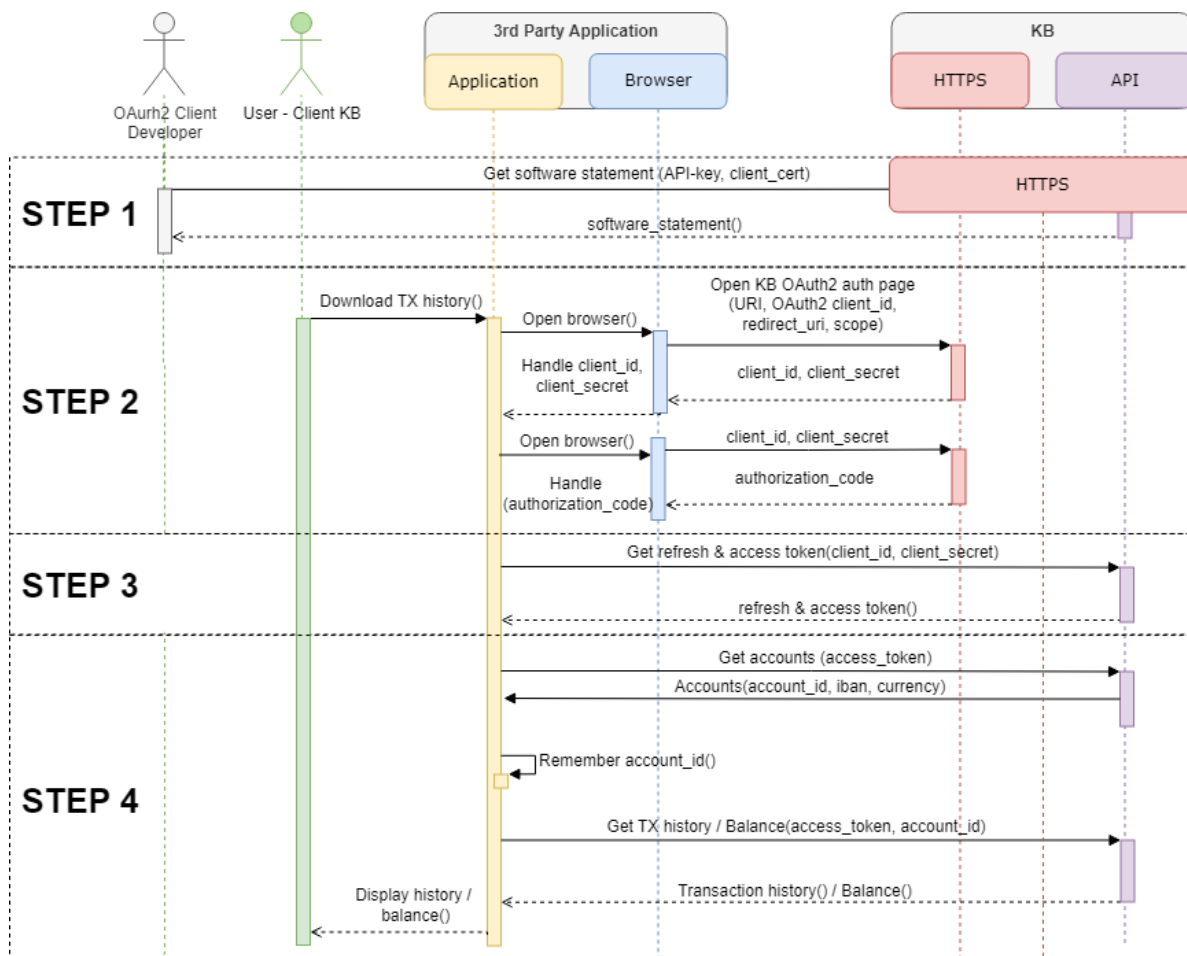
We shall be pleased to answer your questions at [api@kb.cz](mailto:api@kb.cz).

[Postman collection](#) for Sandbox. For more examples see <https://github.com/komercka/adaa-client/wiki/Sandbox>.

## SANDBOX REQUIREMENTS

- **API key** for access to SANDBOX versions of APIs; generate API key and **subscribe** API (**OAuth2 Software Statement Sandbox API, OAuth2 Sandbox API and Account Direct Access Sandbox API**); for more information see [Sandbox API manual](#)
- **Authorization code** to obtain an access token (authorization code is generated by **KB Test login page, see below - STEP 2**)
- **Access token** for ADAA service API (obtain access token by **OAuth2 Sandbox API, see below - STEP 3**)

## HOW DOES IT WORK?







**DECRYPT RESPONSE**Encryption algorithm; **AES-256 (AES/GCM/NoPadding)** .

```
{
  "application_type": "web",
  "redirect_uris": [
    "https://client.example.org/callback-backup",
    "https://client.example.org/callback" ],
  "client_name": "ExampleClient",
  "client_name#en-US": "ExampleClient",
  "logo_uri": "https://client.example.org/logo.png",
  "policy_uri": "https://client.example.org/policy",
  "tos_uri": "https://client.example.org/tos",
  "contact": "example@example.com",
  "contacts": [
    "example@example.com" ],
  "scopes": [
    "adaa" ],
  "scope": "adaa",
  "response_types": [
    "code" ],
  "response_type": "code",
  "grant_types": [
    "refresh_token",
    "authorization_code" ],
  "subject_type": "public",
  "token_endpoint_auth_method": "client_secret_basic",
  "require_auth_time": false,
  "bin": "123456",
  "client_id": "ExampleClient-6303",
  "client_secret": "bUfDQ1fMmfaSlZBZXlxBOQ",
  "api_key": "NOT_PROVIDED",
  "registration_client_uri": "https://caas.kb.cz/openam/json/caas/api/oauth2/register/ExampleClient-6303",
  "client_id_issued_at": 1649407196}
```

**STEP 3 AUTHORIZATION CODE**

- open KB test login page (TPP app must be OAUTH2 compatible)
- enter random client's name, authorization code will be sent to `redirect_uri`

**KB OAUTH2 TEST URI:**

```
https://api.kb.cz/open/api/sandbox/oauth2-authorization-ui/v2/?response_type=code&client_id=client&redirect_uri=REDIRECT_URI&scope=adaa&state=xyz
```

URI parameter	Description
<code>response_type</code>	Type of return value, always fill „code“.
<code>client_id</code>	OAuth2 agent ID – TPP application name (choose any value).
<code>redirect_uri</code>	Back URL for sending an authorization code (i.e. „https://mycompany.cz“).
<code>scope</code>	Authority scope, for ADAA always fill „adaa“.
<code>state</code>	Used for Cross-site request forgery prevention (choose any value).

**Note:** Unsupported parameters route to bad request. Parameter `state` will be sent back to `redirect_uri`.

**KB OAUTH2 TEST login page:**

### Client authentication – test version

**Note:** This is test page for sandbox API. Authorization code is generated from entered data (name of test client).  
Use authorization code in related API services (see sandbox manual of related API services).

The authorization code will be sent to your `redirect_uri`; auth. code will be sent in `code` parameter.

```
https://mycompany.cz?code= eyJ1c2VySWQiOiJLbGllbnQgMSIsInNjb3BlcyI6WyJlaWQiXX0
```

**STEP 4 TOKENS**

- use OAUTH2 SANDBOX API to obtain an access token
- save access token for ADAA SANDBOX API

OAUTH2 SANDBOX API Request parameters, endpoint ***/access-tokens***

URI parameter	Description
<u>code</u>	Received authorization code, see STEP 2.
<u>client_id</u>	OAuth2 agent ID – TPP application name (choose any value).
<u>client_secret</u>	OAuth2 agent ID password – for SANDBOX fill „ <b>password</b> ”.
<u>grant_type</u>	Code type, fill „ <b>authorization_code</b> ”.

You will receive access token in response. Sandbox access\_token is valid for 1 hour (3600 seconds).

#### STEP 5 DATA FROM ACCOUNT

Use ACCOUNT DIRECT ACCES SANDBOX API for test data:

- a) use endpoint accounts to get ID for your accounts,
- b) use endpoint History/Balances with account\_id.

---

#### ACCOUNTS

Before you can obtain transaction history or account balance, you need to obtain account ID for requested IBAN account. For this purpose, you need to use endpoint **account**.

You can store Account ID

#### Request parameters:

URI parameter	Description
api_key	API key generated by API Portal, for more information see <a href="#">Sandbox API manual</a>
access_token	Received authorization code, see STEP 3

**TRANSACTION HISTORY**

If you have account ID, you are able to get transaction history.

**Request parameters:**

URI parameter	Description
api_key	API key generated by API Portal, for more information see <a href="#">Sandbox API manual</a>
access_token	Received authorization code, see STEP 3
account_id	Depend on this parameter you get different response. See table below.

**Sandbox values:**

IBAN and CURRENCY represented as account_id (response from resorce account-ids)	HTTP Response	Description
CZ1301000901147771800227 CZK	200	History for IBAN CZ1301000901147771800227
CZ6101000900930790180287 CZK	200	History for IBAN CZ6101000900930790180287
CZ4401000901148096130227 CZK	200	History for IBAN CZ4401000901148096130227
CZ9601000901148066540217 CZK	200	History for IBAN CZ9601000901148066540217
CZ6801000901147826700267 CZK	200	History for IBAN CZ6801000901147826700267
CZ1150516924342117676733 CZK	403	Error code: ACCOUNT_IN_BLACKLIST
SK7556422734758296432987 EUR	403	Error code: ACCOUNT_ACCESS_DENIED
CZ7630300000001023556014 CZK Other IBANs	404	Error code: ACCOUNT_NOT_FOUND

## BALANCES

If you have account ID, you are able to get account balance.

### Request parameters:

URI parameter	Description
api_key	API key generated by API Portal, for more information see <a href="#">Sandbox API manual</a>
access_token	Received authorization code, see STEP 3
account_id	Depend on this parameter you get different response. See table below.

## ACCOUNT STATEMENT - METADATA:

If you have account id, you are able to get list of **statement ids** (id of specific PDF file) belongs to specific account.

### Request parameters:

URI parameter	Description
api_key	API key generated by API Portal, for more information see <a href="#">Sandbox API manual</a>
access_token	Received authorization code, see STEP 3
account_id	Depend on this parameter you get different response. See table below.

### Sandbox values:

Same as Transaction history.

## ACCOUNT STATEMENT – PDF FILE:

If you have statement Id, you are able to get specific PDF file (statement).

### Request parameters:

URI parameter	Description
api_key	API key generated by API Portal, for more information see <a href="#">Sandbox API manual</a>
access_token	Received authorization code, see STEP 3
account_id	Depend on this parameter you get different response. See table below.
<b>statement_id</b>	ID of specific PDF file.





```
--data-raw '{
  "eventApiUrl": "https://client.example.org",
  "eventApiKey": "no_valid"
}'
```

```
{
  "eventApiUrl": "https://client.example.org",
  "eventApiKey":
}
```

### Response

```
{
  "subscriptionId": "18fa17b0-3c9a-4391-b1cc-dcae92651647",
  "eventApiUrl": "https://client.example",
  "eventApiVersion": "1.0",
  "status": "ACTIVE"
}
```

---

## YOUR API TO RECEIVE NOTIFICATIONS

Objective	Notification Subscription API
Type	REST API service
Called API	[POST], <a href="https://company.org/event-api/v1">https://company.org/event-api/v1</a>
Requirements	The structure of the API corresponds to the example of github
Request data	subscriptionId (id assigned within the registration for subscription of notifications), eventCount (number of changes)
Response data	Information on receipt of notification.

The endpoint is used to receive a notification when there is a change in the account (new event in the transaction history). The API issues a notification to the recipient (third party). The main benefit of this solution is that there is no need to regularly download the transaction history to identify the change. The download can be performed only at the moment of obtaining the notification from KB.

You can see example <https://github.com/komercka/adaa-eventapi-spring-boot-example>