SLOVAK
BANKING
ASSOCIATION

Standard for Push Payment Notification

Version: 1.1

Bratislava, 2025-07-17

License grant

Slovak Banking association (hereinafter referred as *„SBA“*) grants to any contributor, developer, implementer, or other interested party of Standard for Push Payment Notification (hereinafter referred as *„Standard“*) non-exclusive, royalty free, worldwide copyright license to reproduce, prepare derivative works from, distribute, perform and display, this *„Standard“* solely for the purposes of developing and implementing relevant specification and applications.

Provided that attribution be made to *„SBA“* as the source of the material, but that such attribution does not indicate an endorsement by *„SBA“*.

Disclaimer of warranties and limitation of liability

Permission to use the *„Standard“* is hereby granted under the following conditions:

- that *„SBA“* nor contributors to the *„Standard“* shall have any responsibility or liability whatsoever to any other party from the use or publication of the „Standard“;
- that one cannot rely on the accuracy or finality of the *„Standard“*; and
- that the willingness of *„SBA“* to provide the *„Standard“* does not in any way convey or imply any a responsibility for any product or service developed in accordance with the *„Standard“* and *„SBA“* as well as the contributors to the *„Standard“* specifically disclaim any such responsibility to any party.

Implementation of certain elements of this *„Standard“* may require licenses under third party intellectual property rights, including without limitation, patent rights. „SBA“ and any other contributors to the *„Standard“* are not, and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights.

This *„Standard“* is provided "as is", "where is" and "with all faults", and *„SBA“* does not makes any representation or warranty of any kind, express or implied, including any implied warranties of merchantability, non-infringement of third party intellectual property rights (whether or not third parties have been advised, have reason to know, or are otherwise in fact aware of any information), and fitness for a particular purpose (including any errors and omissions in the *„Standard“*).

To the extent permitted by applicable law, neither *„SBA“* nor any contributors to the *„Standard“* shall be liable to any user of the *„Standard“* for any damages (other than direct actual out-of-pocket damages) under any theory of law, including, without limitation, any special damages, nor any damages for loss of business profits, business interruption, loss of business information, or other monetary loss, nor any damages arising out of third party claims (including claims of intellectual property infringement) arising out of the use of or inability to use the *„Standard“*, even if advised of the possibility of such damages.

Document version and history

Version	Release date	Note/ Changes
1.0	2025-06-23	First release of the document.
1.1	2025-07-17	Adding nes mandatory attribute (dataIntegrityHash)and two optional attributes (creditorAccount, creditorName). Errata1: Correction of the date format used in examples: page 11 (2025-07-25) Errata2: Clarification of the Amount field description - rules for the use of decimal places and decimal separator: page 12 (2025-12-04)

Versioning of this document

A normal version number of this document have to take the form X.Y where X represents a Major version and Y a Minor version of this document. Elements X and Y are non-negative integers. Each element have to increase numerically.

Once a versioned document has been released, the contents of that version may not be modified. Any modifications have to be released as a new version.

Version 1.0 defines a final document. The way in which the version number is incremented after this release is dependent on its changes.

Major version have to be incremented if the document has encountered significant and incompatibile changes of specification.

Minor version is incremented if new information is introduced to the document or if information is removed from the document (e.g. errata, errors in specifications whithout affecting the comapatibility of the use of Major version)

Notational conventions

The key words “must”, “must not”, “required”, “shall”, “shall not”, “should”, “should not”, “recommended”, “may”, and “optional” in this document are to be interpreted as described in RFC2119.

Table of content

Foreword

The Slovak Banking Association (hereinafter referred as *„SBA“*) is a key association in Slovakia’s financial sector and the sole organisation representing banks’ interests in the Slovak republic. One of the association’s activity is the development and promotion of common technical standards in the Slovakia’s financial sector.

The work of preparing common technical standards is normally carried out through the special working groups. Each association’s member has the right to participate on the activities of special working group. In general *SBA*’s standards are voluntary for its members. Participation in the development of the association’s technical standards does not imply an obligation of association’s members to implement them.

Common technical standards developed by *SBA* are usually opened and free to use. After the approval, each common technical standard is published on the association’s websites (e. g. www.sbaonline.sk).

1 Introduction

In certain situations, it can be beneficial for bank clients (PSU) to automatically process information about incoming funds to their own accounts. PSU can use this service, for example, to handle instant payments initiated through methods such as QR codes or payment links, which are commonly used in e-commerce or at physical points of sale. This document describes a simple API service (Push Payment Notification) that enables clients to automatically process notifications of incoming payments.

The API service relies on the existence of a technical partner of PSU (TPI), who ensures the operation of the API server in accordance with this standard. Based on mutual agreements with the PSU, the ASPSP (bank) sends information about incoming funds to this server.

SBA has prepared this document in an effort to contribute to the development of innovative payment solutions based on the instant payment scheme. The provision of this service by banks is voluntary, and the publication of this standard does not obligate SBA members to offer such a service.

1.1 Document purpose

Standard for Push Payment Notification (hereinafter referred as „*Standard*") provides the information how to implement the API service for Push Payment Notification for any developer, implementer, or other interested party.

The „*Standard*" specifically explains:

- proces flow of notification,
- secure communication,
- request call and responses for API service.

1.2 Character sets

The character set is UTF 8 encoded. For strings at least the following character set will be accepted by participants:

a b c d e f g h i j k l m n o p q r s t u v w x y z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

0 1 2 3 4 5 6 7 8 9

/ - ? : ( ) . , ' +

Space

Character	ASCII	Unicode
/	47	U+002F
-	45	U+002D
?	63	U+003F
:	58	U+003A
(	40	U+0028
)	41	U+0029

Character	ASCII	Unicode
.	46	U+002E
,	44	U+002C
'	39	U+0027
+	43	U+002B
Space	32	U+0020

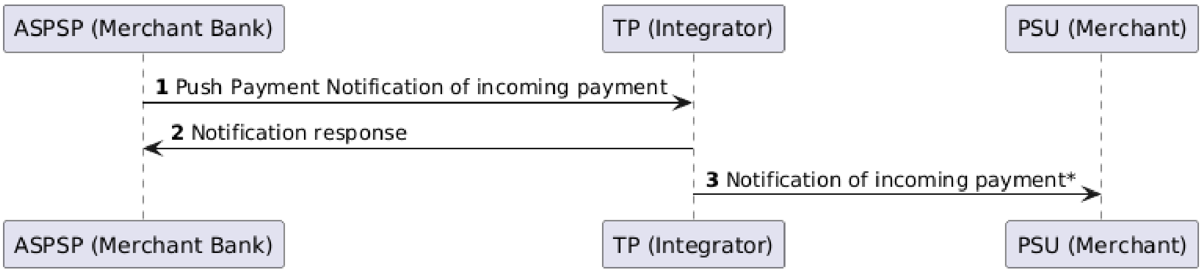
1.3 Terms and definition

For the purposes of this document, the following terms and definition aply.

Term	Meaning
ASPSP	Account Servicing Payment Service Provider in accordance with PSD2 (e.g. Merchant Bank).
eIDAS	Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC
IBAN	International Bank Account Number defined by ISO 13616.
ISO 20022	an universal financial industry message scheme.
PSD2	Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC
PSU (Merchant)	Payment Service User in accordance with PSD2; (e.g. Merchant)
SBA	Slovak Banking Association; a key association in Slovakia’s financial sector and the sole organisation representing banks’ interests in Slovakia.
SEPA	Single European Paymnet Area; a payment-integration initiative of the European Union for simplification of bank transfers denominated in euro.
SSL	Secure Sockets Layer (SSL) is a security protocol that provides privacy, authentication, and data integrity for Internet communications
TLS	Transport Layer Security (TLS) is a cryptographic protocol designed to provide secure communications over a computer network
TPI	Third party - Integrator is the technical partner of PSU, which ensures communication between ASPSP and PSU. TPI provides the API interface in accordance with this document.

2 Proces flow

The following flow shows the simple request and response flow for a resource status notification service:



* Notification of incoming payment from TPI to PSU (Merchant) is out of the scope of this document.

1. Once the Merchant Bank receives an instant payment in the Merchant account, it generates a Push payment notification for the incoming payment and sends it to the TP Integrator API, as described in section 4.1 RequestResponse,
2. The TP Integrator processes the incoming Push payment notification and responds with a "Notification response," as detailed in section 4.2 Response,
3. The TP Integrator then forwards the Push payment notification to the Merchant (e.g., directly to the merchant's cash register).

3 Security

The communication is always secured by using a TLS connection using TLS version 1.2 or higher.

For securing API calls between the TPI (Server) and the ASPSP (Client), mutual SSL authentication (or 2-way authentication) is preferred. Both the ASPSP and TPI present their respective digital certificates during the SSL/TLS handshake process.

The certificates for mutual SSL authentication should be issued by qualified trust service provider in accordance with the eIDAS Regulation. Trusted list of qualified trust service providers is provided by the Member States of the European Union and European Economic Area and is published on the EU website.

ASPSP usually uses eIDAS Qualified Website Authentication Certificates (QWAC). This type of certificate may be also use for mutual SSL authentication. TPI has several options to recognize ASPSP based on the certificate (e.g. based on domain, bussiness name, licence number, identification number).

In mutual SSL authentication, TPI presents itself to APSPS using an Organization Validation Certificate. It is strongly recommended that TPI implements a process for managing digital certificates of ASPSPs.

The API interface on the TPI side is dentified by Domain Name System (DNS) and not by IP address.

4 Service: Push Payment Notification

4.1 Request

Endpoint: POST <TPI-Notification-URI>

4.1.1 Parameters

No Path and No Query Parameters

4.1.2 Request header

Attribute	Type	Condition	Description
Content-Type	String	Mandatory	The string application/json is used.
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, determined by the initiating party (ASPSP).
Date	ISODateTime	Mandatory	Timestamp of message call. Recommendation to use only: UTC or UTC offset format.

4.1.3 Request body

Attribute	Type	Condition	Description
transactionStatus	Transaction Status (code lists)	Mandatory	Transaction code based on ISO20022 External code sets (e.g. ACCC). (Please see Annex A).
transactionAmount	Amount (complex data type)	Mandatory	The Amount of the transaction with Currency (Please see Annex A).
endToEndId	Max. 35 Text	Mandatory	A unique identifier created by the TP-Integrator (e.g.Transaction_ID of cash receipt).
dataIntegrityHash	Max. 64 Hexadecimal	Mandatory	The value of this attribute represents a hash in hexadecimal string format, calculated using the SHA-256 algorithm from concatenated data: iban (from creditorAccount attribute), amount and currency code (from transactionAmount attribute) and endToEndId attribute. (Please see Annex B)
creditorAccount	Account reference (complex data type)	Optional	Attribute refers to the account to which the payment will be credited in the form of IBAN. (please see Annex A)
creditorName	Max. 70 Text	Optional	Attribute specifies the name of the individual or entity receiving the payment.

4.2 Response

4.2.1 HTTP Response code:

200

4.2.2 Response Header

Attribute	Type	Condition	Description
Content-Type	String	Mandatory	The string application/json is used.
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, determined by the initiating party (ASPSP).
Date	ISODateTime	Mandatory	Timestamp of response message call. Recommendation to use only: UTC or UTC offset format.

4.3 Error codes

No additional error information is provided for this simple service.

Status Code	Error Code	Description
400	Bad Request	Validation error occurred. This code will cover malformed syntax in request or incorrect data in payload.
401	Unauthorized	The TPI or the PSU is not correctly authorized to perform the request. Retry the request with correct authentication information.
403	Forbiden	Returned if the resource that was referenced in the path exists but cannot be accessed by the ASPSP. This code should only be used for non-sensitive id references as it will reveal that the resource exists even though it cannot be accessed.
404	Not Found	Returned if the endpoint that was referenced in the path does not exist or cannot be referenced by the ASPSP. When in doubt if a specific id in the path is sensitive or not, use the HTTP response code 404 instead of the HTTP response code 403.
405	Method Not Allowed	This code is only sent when the HTTP method (PUT, POST, DELETE, GET etc.) is not supported on a specific endpoint.
408	Request Timeout	The server is still working correctly, but an individual request has timed out.
415	Unsupported Media Type	The ASPSP has supplied a media type which the TPI does not support.
500	Internal Server Error	Internal server error occurred.
503	Service Unavailable	The TPP server is currently unavailable. Generally, this is a temporary state.

4.4 Example

4.4.1 Request

4.4.1.1 Header:

```
POST <TPI-Notification-URI>
Content-Type: application/json
X-Request-ID: 6478e8f0-71e6-478a-a609-494865868457
Date: 2025-05-28T00:20:00Z
```

4.4.1.2 Payload:

```
{
"transactionStatus": "ACCC"
"endToEndId": "QR-ab29e346f1d841c8a95a63d857490818",
"transactionAmount": {"currency": "EUR", "amount": "123.45"},
"dataIntegrityHash":"b150d2343fefd404f89788efece5e0c6bd423005553d708fb40bf600b1f
4c8ae",
"creditorAccount": {"iban": "SK4811000000002944116480"},
"creditorName":"Merchant Name, sro"
}
```

4.4.2 Response

4.4.2.1 Header:

```
HTTP/1.x 200
Content-Type: application/json
X-Request-ID: 6478e8f0-71e6-478a-a609-494865868457
Date: 2025-05-28T00:20:00Z
```

Annexes

A. Complex Data Types and Code Lists

Transaction status

Based on External Code Set of ISO 20022 (ExternalPaymentTransactionStatus1Code).

The ACCC (AcceptedSettlementCompletedCreditorAccount[1]) code is supported.

Amount

Attribute	Type	Condition	Description
currency	Currency Code	Mandatory	ISO 4217 Alpha 3 currency code (EUR)
amount	String	Mandatory	The value must be a decimal number using a dot (".") as the decimal separator. The integer part may be either zero or a non-zero number without leading zeros, containing up to nine digits. Exactly two digits must follow the decimal separator.

Account Reference

Attribute	Type	Condition	Description
iban	IBAN	Mandatory	International Bank Account Number (ISO 13616) without whitespaces.

[1] AcceptedSettlementCompletedCreditorAccount (ACCC): Settlement on the creditor's account has been completed.

B. How to generate “dataIntegrityHash” attribute

The value of this attribute represents a hash in hexadecimal string format, calculated using the SHA-256 algorithm from concatenated data: iban, amount, currency, and endtoEndId. The data is concatenated in the following order: IBAN | amount | currency | endToEndId, using the character **|** (vertical bar, or pipe[2]) as a separator.

Input parameters:

- iban: value from creditorAccount attribute without whitespaces; ensure that UPPERCASE letters are always used. [SK4811000000002944116480]
- amount: value from transactionAmount attribute; ensure that decimal numbers are always used, with a dot “.” as the decimal separator**.** The number of digits following the decimal separator is determined by the “minor unit“ as specified in ISO 4217[3]; e.g. for EUR currency always use 2 digits after the decimal separtor [123.45], [12345.00] or [0.12]
- currency: value from transactionAmount attribute [EUR]
- endToEndId: value of endToEndId attribute [QR-ab29e346f1d841c8a95a63d857490818]

Output:

- lowercase hash, 32 bytes

Pseudocode of hash function

```
function generateDataIntegrityHash(iban, amount, currency, endToEndId):
      # Step 1: Concatenate the input strings with "|" delimiter (pipe)
      inputString = iban + "|" + amount + "|" + currency + "|" + endToEndId
      # Step 2: Compute SHA-256 hash of the concatenated string
      sha256Hash = SHA256(inputString)
      # Step 3: Convert the SHA-256 hash (binary) to a hexadecimal string
      hashHex = toHex(sha256Hash)
      # Step 4: Return the hexadecimal hash string
      RETURN hashHex
```

Example

```
generateTransactionHash ("SK4811000000002944116480","123.45","EUR","QR-
ab29e346f1d841c8a95a63d857490818")
      inputString="SK4811000000002944116480|123.45|EUR|QR-
ab29e346f1d841c8a95a63d857490818"
      hashHex="b150d2343fefd404f89788efece5e0c6bd423005553d708fb40bf600b1f4c8ae"
      dataIntegrityHash = hashHex
```

[2] ASCII code 124, Unicode U+007C

[3] ISO 4217 Currency Codes

Bibliography

[1] Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC (PSD2),

[2] EU/EEA Trusted List Browser,

[3] ISO 4217 – Currency codes,

[4] ISO 13616 – International bank account number (IBAN),

[5] ISO 20022 – Universal financial industry message scheme,

[6] ISO 20022 – External code sets | ISO20022,

[7] NextGenPSD2 Framework Implementation Guidelines – Extended Services, Resource Status Notification Service,

[8] NextGenPSD2 Framework, Implementation Guidelines – Joint Initiative on a PSD2 Compliant XS2A Interface,

[9] Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC,

[10] RFC 5987 – Character Set and Language Encoding for HTTP Header Field Parameters,

[11] RFC 5246 – The Transport Layer Security (TLS) Protocol,

[12] RFC 3986 – Uniform Resource Identifiers (URI): Generic Syntax.

Slovak banking association, www.sbaonline.sk, sba@sbaonline.sk