                 The 'payto' URI scheme for payments
                         draft-dold-payto-03

Abstract

   This document defines the 'payto' Uniform Resource Identifier (URI)
   scheme for designating targets for payments.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This document defines the 'payto' Uniform Resource Identifier (URI)
[RFC3986] scheme for designating targets for payments. In its
simplest form, a 'payto' URL identifies a payment target type and
optionally a target identifier. Additional parameters, such as an
amount or a payment reference, can be provided.

The interpretation of the target identifier is defined by the payment
target type, and typically represents either a bank account or an
(unsettled) transaction.

2. Syntax of a 'payto' URL

This document uses the Augmented Backus-Naur Form (ABNF) of
[RFC5234].

```
payto-URI = "payto" "://" authority path-abempty [ "?" opts ]
opts = opt *( "&amp;" opt )
opt = (generic-opt / authority-specific-opt) "=" *( pchar )
generic-opt = "amount" / "creditor-name" / "debitor-name" /
              "message" / "instruction"
authority = <authority, see [RFC3986], Section 3.2>
path-abempty = <path-abempty, see [RFC3986], Section 3.3>
pchar = <pchar, see [RFC3986], Appendix A.>
```

3. Semantics

   The authority component of a payment URI identifies the payment
   target type. The payment target types are defined in the Payto
   Payment Target Type Registry, see Section 8.2. The path component of
   the URI identifies the target for a payment as interpreted by the
   respective payment target type. The query component of the URI can
   provide additional parameters for a payment. Every payment method
   SHOULD accept the options defined in generic-opt. The default
   operation of applications that invoke a URI with the payto scheme
   SHOULD be to launch an application (if available) associated with the
   payment target type that can initiate a payment. If multiple
   handlers are registered for the same payment target type, the user
   SHOULD be able to choose which application to launch. This allows
   users with multiple bank accounts (each accessed the respective
   bank's banking application) to choose which account to pay with.
   Details of the payment MUST be taken from the path and options given
   in the URI. The user SHOULD be allowed to modify these details
   before confirming a payment.

4. Examples

      payto://sepa/CH9300762011623852957?amount=EUR:200.0&message=hello

      INVALID (authority missing): payto:sepa/12345

5. Generic Options

   Applications MUST accept URIs with options in any order. The
   "amount" option MUST only occur at most once. Other options MAY be
   allowed multiple times, with further restrictions depending on the
   payment method. The following options SHOULD be understood by every
   payment method.

   amount: The amount to transfer, including currency information if
   applicable. The format MUST be:

```
     amount = [ currency ":" ] unit [ "." fraction ]
     currency = 1*ALPHA
     unit = 1*(DIGIT / ",")
     fraction = 1*(DIGIT / ",")
```

   The unit value MUST be smaller than 2^53. If present, the fraction
   MUST consist of no more than 8 decimal digits. The use of commas is
   optional for readability and they MUST be ignored.

creditor-name: Name of the entity that is credited (receives the payment).

debitor-name: Name of the entity that is debited (makes the payment).

message: A short message to identify the purpose of the payment, which MAY be subject to lossy conversions (for example, due to character set encoding limitations).

instruction: A short message giving instructions to the recipient, which MUST NOT be subject to lossy conversions. Character set limitations allowed for such instructions depend on the payment method.

6. Internationalization and Character Encoding

Various payment systems use restricted character sets. An application that processes 'payto' URIs MUST convert characters that are not allowed by the respective payment systems into allowable character using either an encoding or a replacement table. This conversion process MAY be lossy, except for the instruction field.

To avoid special encoding rules for the payment target identifier, the userinfo component [RFC3986] is disallowed in payto URIs. Instead, the payment target identifier is given as an option, where encoding rules are uniform for all options.

7. Security Considerations

Interactive applications handling the payto URI scheme MUST NOT initiate any financial transactions without prior review and confirmation from the user, and MUST take measures to prevent clickjacking [HMW12].

Unless a payto URI is received over a trusted, authenticated channel, a user might not be able to identify the target of a payment. In particular due to homographs [unicode-tr36], a payment target type SHOULD NOT use human-readable names in combination with unicode in the target account specification, as it could give the user the illusion of being able to identify the target account from the URL.

To avoid unnecessary data collection, payment target types SHOULD NOT include personally identifying information about the sender of a payment that is not essential for an application to conduct a payment.

8. IANA Considerations

8.1. URI Scheme Registration

   The "payto" URI scheme is to be registered in the "Permanent URI
   Schemes" registry.

      Scheme name: payto

      Status: permanent

      URI scheme syntax: See Section 2.

      URI scheme semantics: See Section 3.

      Applications/protocols that use this scheme name: payto URIs are
      mainly used by financial software

      Contact: grothoff@gnu.org

      Change controller: grothoff@gnu.org

      References: See References section of this document.

8.2. Payment Target Type Registry

   This document defines a registry for payment methods. The name of
   the registry is "Payment Target Type Registry".

   The registry shall record for each entry:

   o Name: The name of the payment target type (case insensitive ASCII
      string)

   o Description: A description of the payment target type, including
      the semantics of the path in the URI if applicable.

   o Contact: The contact information of a person to contact for
      further information

   o References: Optionally, references describing the payment method
      (such as an RFC) and method-specific options

   The registration policy for this registry is "First Come First
   Served", as described in [RFC5226].

   The registry is initially populated with the following entries:

Name	Description	Contact>	References
ach	Automated Clearing House. The path is a bank account number.	N/A	[NACHA]
sepa	Single European Payment Area. The path is an IBAN.	N/A	[ISO20022]
upi	Unified Payment Interface. The path is an account alias.	N/A	[UPILinking]
bitcoin	Bitcoin protocol. The path is a "bitcoinaddress" as per [BIP0021].	N/A	[BIP0021]
ilp	Interledger protocol. The path is an ILP address as per [ILP-ADDR].	N/A	[ILP-ADDR]

9. References

9.1. Normative References

[ISO20022]
          International Organization for Standardization, "ISO 20022
          Financial Services - Universal financial industry message
          scheme", May 2013.

[NACHA]   NACHA, "NACHA Operating Rules & Guidelines", January 2017.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
          Resource Identifier (URI): Generic Syntax", STD 66,
          RFC 3986, DOI 10.17487/RFC3986, January 2005,
          <https://www.rfc-editor.org/info/rfc3986>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an
          IANA Considerations Section in RFCs", RFC 5226,
          DOI 10.17487/RFC5226, May 2008,
          <https://www.rfc-editor.org/info/rfc5226>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax
          Specifications: ABNF", STD 68, RFC 5234, January 2008.

[unicode-tr36]
          Davis, M., Ed. and M. Suignard, "Unicode Technical Report
          #36: Unicode Security Considerations", September 2014.

9.2. Informational References

   [BIP0021]   Schneider, N. and M. Corallo, "Bitcoin Improvement
               Proposal 21", January 2012,
               <https://en.bitcoin.it/wiki/BIP_0021>.

   [HMW12]   Huang, L., Moshchuk, A., Wang, H., Schecter, S., and C.
               Jackson, "Clickjacking: Attacks and Defenses", January
               2012, <https://www.usenix.org/system/files/conference/
               usenixsecurity12/sec12-final39.pdf>.

   [ILP-ADDR]
               Interledger Team, "ILP Addresses - v2.0.0", September
               2018, <https://interledger.org/rfcs/0015-ilp-addresses/>.

   [UPILinking]
               National Payment Corporation of India, "Unified Payment
               Interface - Common URL Specifications For Deep Linking And
               Proximity Integration", May 2016,
               <http://www.npci.org.in/documents/
               UPILinkingSpecificationsVersion10draft.pdf>.

Authors' Addresses

   Florian Dold
   Taler Systems SA
   7, rue de Mondorf
   Erpeldange L-5421
   LU

   Email: dold@taler.net

   Christian Grothoff
   BFH
   Hoeheweg 80
   Biel/Bienne CH-2501
   CH

   Email: christian.grothoff@bfh.ch