The 'payto' URI scheme for payments

draft-dold-payto-00

Abstract

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

Status of This Memo

This Internet-Draft is submitted in full conformance with the
provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering
Task Force (IETF). Note that other groups may also distribute
working documents as Internet-Drafts. The list of current Internet-
Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months
and may be updated, replaced, or obsoleted by other documents at any
time. It is inappropriate to use Internet-Drafts as reference
material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 30, 2017.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the
document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal
Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info) in effect on the date of
publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.
1. Introduction

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

2. Scheme Syntax

This document uses the Augmented Backus-Naur Form (ABNF) of [RFC5234]. Certain values are included by reference from [RFC3986]:

```
payto-URI = "payto" ":/" method account [ "?" params ]
params = param *( "&" param )
param = (generic-param / method-specific-param) ":" *pchar
generic-param = "amount" / "recipient-name" / "sender-name" /
    "message" / "instruction"
method-specific-param = *pchar
method = <authority, see [RFC3986], Section 3.2>
account = <path-abempty, see [RFC3986], Section 3.3>
pchar = <pchar, see [RFC3986], Appendix A.>
amount = [ currency "":" ] unit [ "." fraction ]
currency = 1*ALPHA
unit = 1*(DIGIT / ",")
fraction = 1*(DIGIT / ",")
```
3. Scheme Semantics

The authority component of a payment URI identifies the payment method. The registry for payment methods is defined in Section 5 of this document. The path component of the URI identifies the target account for a payment as interpreted by the respective payment method. The query component of the URI can provide additional parameters for a payment. Every payment method SHOULD accept the options defined in generic-opt.

4. Examples

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

   INVALID (authority missing): payto:sepa/12345

5. Payment Methods

   sepa: Single European Payment Area. The path is an IBAN, as defined by [ISO20022].

   upi: Unified Payment Interface. The path is an account alias, as defined by [UPILinking].

   bitcoin: Bitcoin protocol. The path is a bitcoinaddress, as defined by [BIP0021].

   ach: Automated Clearing House. The path is a bank account number, as defined by [NACHA]

6. Generic Options

   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 fraction MUST be smaller than 10^8. The unit value MUST be smaller than 2^53. The use of commas is optional for readability and they MUST be ignored.

---

recipient-name: Name of the recipient of the payment.

sender-name: Name of the sender of 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 instructions depend on the payment method.

7. Interoperability

This URI scheme subsumes other URI schemes for payments, such as "bitcoin" (provisionally registered) and "upi" (unregistered but in use, see [UPILinking]), providing one unified scheme for payment information.

8. Encoding

The payto URI scheme encoding conforms to the encoding rules established for URIs in [RFC3986]. 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 is typically not lossless.
9. Security Considerations

The payto URL contains instructions on how to send money. Applications that support the payto URI scheme MUST ask for confirmation from the user in order to confirm a payment. Applications MUST handle payto URLs in conformance with the principle of safe interaction (http://www.w3.org/TR/webarch/#safe-interaction).

10. References

10.1. Normative References


10.2. Informational References


Authors' Addresses