Internet Open Trading Protocol (IOTP) HTTP Supplement

<draft-ietf-trade-iotp-http-07.txt>

Donald E. Eastlake 3rd
Chris J. Smith

Status of This Document

Distribution of this document is unlimited. Comments should be sent to the TRADE WG mailing list <ietf-trade@lists.eListX.com> or to the authors.

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months. Internet-Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet-Drafts as reference material or to cite them other than as a ``working draft'' or ``work in progress.''

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

Internet Open Trading Protocol (IOTP [RFC 2801]) messages will be carried as XML documents. As such, the goal of mapping to the transport layer is to ensure that the underlying XML documents are carried successfully between the various parties.

This documents describes that mapping for the Hyper Text Transport Protocol (HTTP), Versions 1.0 and 1.1.

Table of Contents

1. Introduction

Internet Open Trading Protocol (IOTP) messages will be carried as XML [XML] documents. As such, the goal of mapping to the transport layer is to ensure that the underlying XML documents are carried successfully between the various parties.

This documents describes that mapping for the Hyper Text Transport Protocol (HTTP), Versions 1.0 and 1.1 [RFCs 1945, 2616].

There may be future documents describing IOTP over email (SMTP), TCP, cable TV, or other transports.

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].

2. HTTP Servers and Clients

The structure of IOTP maps on to the structure of HTTP in the following way:

- The merchant, payment handler, delivery handler, and customer care roles are all represented by HTTP servers. Each may be represented by a separate server, or they may be combined in any combination.

- The consumer role is represented by an HTTP client.

Note: A Merchant, may act in the role of a consumer, for example to deposit electronic cash. In this case the Merchant, as an organization rather than as a role, would need to be supported by an HTTP client.

3. HTTP Net Locations

The Net Locations contained within the IOTP specification are all URIs [RFC 2396]. If a secure connection is required or desired a secure channel that both the HTTP Server and Client support MUST be used. Examples of such channels are SSL version 3 or TLS [RFC 2246].

4. Consumer Clients

In most environments, the consumer agent will initially be an HTML browser. However, current browsers do not provide the needed

capability to act as an agent for the consumer for an IOTP transaction. This leads to two requirements:

a method of starting and passing control to the IOTP client, and

a method of closing down the IOTP client cleanly and passing control back to the HTML browser once the IOTP Transaction has finished.

4.1 Starting the IOTP Client and the Merchant IOTP Server

At some point, the HTTP client at the consumer will send an HTTP request that is interpreted as an "IOTP Startup Request" by the Merchant HTTP server. This might, for example, be the result of clicking on a "pay" button. This message is a stand-in for a request message of some form and the Merchant Server will respond with the first IOTP Message in the form of an XML document.

The MIME type for all IOTP messages is: "APPLICATION/IOTP"; however "APPLICATION/X-IOTP" has been in use for experimentation and development and SHOULD also be recognized. See section 7 below for the MIME type registration template for APPLICATION/IOTP. Because HTTP is binary clean, no content-transfer-encoding is required. (See [RFC 2376] re the application/xml type which has some similar considerations.)

This HTTP response will be interpreted by the HTML browser as a request to start the application associated with MIME type "APPLICATION/IOTP", and to pass the content of this message to that application.

At this point, the IOTP client will be started and have the first message.

IOTP messages are short-lived. Therefore, the HTTP server SHOULD avoid having its responses cached. In HTTP V1.0, the "nocache" pragma can be used. This can be neglected on SSL/TLS secured connections which are not cached and on HTTP POST requests in HTTP v1.1 as in v1.1 POST responses are not cached.

4.2 Ongoing IOTP Messages

Data from earlier IOTP Messages in a transaction MUST be retained by the IOTP Client so that it may (1) be copied to make up part of later IOTP messages, (2) used in calculations to verify signatures in later IOTP message, (3) be resent in some cases where a request has timed

out without response, (4) used as input to the Customer Care role in

later versions of IOTP, etc. The way in which the data is copied depends on the IOTP Transaction. The data MUST be retained until the end of the transaction, whether by success, failure, or cancelation, and as long thereafter as it is desired for any of the parties to inquire into it.

The IOTP messages contain Net Locations (e.g. the PayReqNetLocn) which for HTTP will contain the URIs to which the IOTP client MUST send IOTP messages.

Subsequent IOTP messages (XML documents) will be sent using the POST function of HTTP. The HTTP client MUST perform full HTTP POST requests.

The XML documents MUST be sent in a manner compatible with the external encodings allowed by the XML [XML] specification.

4.3 Stopping an IOTP Transaction

The following should be read in conjunction with [RFC 2801].

An IOTP Transaction is complete when

-- the IOTP client decides to fail the IOTP Transaction for some reason either by canceling the transaction or as a result of discovering an error in an IOTP message received, or

-- a "time out" occurs or a connection fails, e.g. a response to an IOTP Message, has not been received after some user-defined period of Time (including retransmissions).

An IOTP Client which processes an IOTP Transaction which:

-- completes successfully (i.e. it has not received an Error Block with a HardError or a Cancel Block) MUST direct the browser to the Net Location specified in SuccessNetLocn in the Protocol Options Component, i.e., cause it to do an HTTP GET with that URL.

-- does not complete successfully, because it has received some Error Trading Block, MUST display the information in the Error Message, stop the transaction, and pass control to the browser so that it will do a GET on the Error Net Location specified for the role from which the error was received.

-- is cancelled since a Cancel Block has been received, MUST stop the IOTP Transaction and hand control to the browser so that it will do a GET on the on the Cancel Net Location specified for the role

from which the Cancel Block was received.

-- is in error because an IOTP Message does not conform to this specification, MUST send an IOTP Message containing a Error Trading Block to role from which the erroneous message was received and the ErrorLogNetLoc specified for that role, stop the IOTP Transaction, and hand control to the browser so that it will do a GET from the Error Net Location specified for the role from which the bad message was received.

-- has a "time out", MUST display a message describing the time out. May give the user the option of cancelling or retrying and/or may automatically retry. On failure due to time out, treat as an error above.

Each implementation of an IOTP client may decide whether or not to terminate the IOTP Client application immediately upon completing an IOTP Transaction or whether to wait until it is closed down as a result of, for example, user shut down or browser shut down.

5. Starting the Payment handler and Deliverer IOTP Servers

Payment Handler and Deliverer IOTP Servers are started by receiving an IOTP Message which contains:

-- for a Payment handler, a Payment Request Block, and

-- for a Delivery Handler, a Delivery Request Block

6. Security Considerations

Security of Internet Open Trade Protocol messages is primarily dependent on signatures within IOTP as described in [RFC 2801] and [RFC 2802]. Privacy protection for IOTP interactions Should can be obtained by using a secure channel for IOTP messages, such as SSL/TLS [RFC 2246].

Note that the security of payment protocols transported by IOTP is the responsibility of those payment protocols, NOT of IOTP.

7. IANA Considerations

This specification defines the APPLICATION/IOTP MIME type. The registration template is as follows [RFC 2048]:

To: ietf-types@iana.org

Subject: Registration of MIME media type APPLICATION/IOTP

MIME media type name: APPLICATION

MIME subtype name: IOTP

Required parameters: (none)

Optional parameters: charset - see RFC 2376

Encoding considerations: Content is XML and may in some cases require quoted printable or base64 encoding. However, no encoding is required for HTTP transport which is expected to be common.

Security considerations: IOTP includes provisions for digital authentication but for confidentiality, other mechanisms such as TLS should be used. See RFC 2801 and RFC 2802.

Interoperability considerations: See RFC 2801.

Published specification: See RFC 2801 and RFC 2802.

Applications which use this media type: Internet Open Trading Protocol applications.

Additional information: (none)

Person & email address to contact for further information:
Name: Donald E. Eastlake 3rd
Email: Donald.Eastlake@motorola.com

Intended usage: COMMON

Author/Change controller: IETF

References

[RFC 1945] - "Hypertext Transfer Protocol -- HTTP/1.0", T. Berners-Lee, R. Fielding & H. Frystyk. May 1996.

[RFC 2048] - "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedure", N. Freed, J. Klensin, J. Postel, November 1996.

[RFC 2119] - "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997.

[RFC 2246] - "The TLS Protocol Version 1.0", T. Dierks, C. Allen. January 1999.

[RFC 2376] - "XML Media Types", E. Whitehead, M. Murata. July 1998.

[RFC 2396] - "Uniform Resource Identifiers (URI): Generic Syntax", T. Berners-Lee, R. Fielding, L. Masinter, August 1998.

[RFC 2616] - "Hypertext Transfer Protocol -- HTTP/1.1", R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, June 1999.

[RFC 2801] - "Internet Open Trading Protocol - IOTP Version 1.0", D. Burdett, April 2000.

[RFC 2802] - "Digital Signatures for the v1.0 Internet Open Trading Protocol (IOTP)", K. Davidson, Y. Kawatsura, April 2000

[XML] - "Extensible Markup Language (XML) 1.0" <http://www.w3.org/TR/REC-xml>, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, 10 February 1998

Authors Addresses

Donald E. Eastlake 3rd
Motorola
140 Forest Avenue
Hudson, MA 01749 USA

Telephone: +1 978-562-2827(h)
+1 508-261-5434(w)
FAX: +1 508-261-4447(w)
email: Donald.Eastlake@motorola.com

Chris J. Smith
Royal Bank of Canada
277 Front Street West
Toronto, Ontario M5V 3A4 CANADA

Telephone: +1 416-348-6090
FAX: +1 416-348-2210
email: chris.smith@royalbank.com

Expiration and File Name

This draft expires December 2000.

Its file name is draft-ietf-trade-iotp-http-07.txt.