XEP-0422: Message Fastening

Kevin Smith
mailto:kevin.smith@isode.com

Dave Cridland
mailto:dave@hellopando.com
xmpp:dwd@dave.cridland.net

2019-12-30
Version 0.2.0

Status	Type	Short Name
Deferred	Standards Track	fasten

This specification defines a way for payloads on a message to be marked as being logically fastened to a previous message.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

There are many situations in which a message contains a payload that adds additional information to the context of a previous message. Examples of this might be a user sending a reaction (e.g. a thumbs-up emoji) to another user's message, or a server adding information on a link previously posted to a chat room, a user updating/editing the content of a message they've previously sent. This specification provides a generic wrapper to fasten such new payloads to the message to which they apply.

2 Discovering support

If a client implements message fastening (that is, it is capable of understanding the payload when received), it MUST specify the 'urn:xmpp:fasten:0' feature in its service discovery information features as specified in Service Discovery (XEP-0030) [1] and the Entity Capabilities profile specified in Entity Capabilities (XEP-0115) [2].

Listing 1: Client requests information about a chat partner's client

```
<iq type='get'
    to='romeo@montague.lit/orchard'
    from='juliet@capulet.lit/balcony'
    id='info1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 2: Partner's client advertises support for message fastening

```
<iq type='result'
    to='juliet@capulet.lit/balcony'
    from='romeo@montague.lit/orchard'
    id='info1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
...
    <feature var='urn:xmpp:fasten:0'/>
...
  </query>
</iq>
```

[1] XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.

[2] XEP-0115: Entity Capabilities <https://xmpp.org/extensions/xep-0115.html>.

3 Use Cases

3.1 Wrapped Payloads

In order to mark that a payload applies to a previous message, a message is sent containing an ”apply-to” element in the namespace ”urn:xmpp:fasten:0”, with attribute ”id” that contains the Unique and Stable Stanza IDs (XEP-0359) [3] origin-id of the stanza to which it applies, the children of which element are those that apply to the previous message (these are ”wrapped payloads” because they are wrapped inside the <apply-to> element). The id of this apply-to-containing message is unimportant, and the type SHOULD be ”normal” (KS: I don't care about this, it just seemed easier to spell it out).
If ”user2@chatservice.example” wanted to send their approval (using a fictional 'i-like-this' element) for a message previously sent to chat room ”chatroom@chatservice.example” with origin-id ”origin-id-1”, they would do the following.
An <apply-to> may contain several fastenings, but they must all be of the same type (the same qualified name)

Listing 3: User2 sends fictional approval

```
<message id=”2” from=”user2@chatservice.example” to ”
   chatroom@chatservice.example”>
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-1”>
      <i-like-this xmlns=’urn:example:like’/>
  </apply-to>
</message>
```

3.2 External Payloads

Where the payload being fastened needs top-level child elements of the stanza holding the fastening (e.g. if a message edit uses the top-level <body> child), these can be listed in <external> child elements of the <apply-to> element, in the same namespace. An <external> element has two attributes: the 'name' attribute MUST be included, and is the name of the top-level stanza child element; the 'element-namespace' attribute specifies the namespace of the element, or if absent indicates that the element is in the same namespace as the stanza (eliding the stream namespace in this way avoids issues when transporting fastenings across combinations of 'jabber:server' and 'jabber:client' streams.
If ”user2@chatservice.example” wanted to send an edit (using a fictional protocol) of a message they previously sent with origin-id 'origin-id-2', and the edit protocol used top-level <body> and <custom> elements, it would look like this.

Listing 4: User2 applies

```
<message id=”4” from=”user2@chatservice.example” to=”
   chatroom@chatservice.example”>
```

[3] XEP-0359: Unique and Stable Stanza IDs <https://xmpp.org/extensions/xep-0359.html>.

```
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-2”>
    <edit xmlns=’urn:example.edit’/>
    <external name=’body’/>
  </apply-to>
  <body>Hi there</body>
  <custom xmlns=’urn:example:custom’>New data</custom>
</message>
```

Note that the first child of the <apply-to> should be the fastening type, and the <external> should follow. If a receiving entity finds unknown namespaced elements inside an <apply-to> with qualified name different to the fastening type, they should be ignored (this allows us to introduce new fastening children in the future, similar to <external>, and maintain backwards compatibility).

3.3 Replacing fastenings

A fastening of a given type (qualified name) can be replaced by sending a new fastening of the same qualified name - as with the original fastening there may be several fastenings of this type (qualified name), but all fastenings must be of the same type. The child payloads of the <apply-to> element then logically replace all fastenings previously applied by the sending entity with the same qualified name (name and namespace).
For example, if user2 wished to update the fastening from the first example they would send:

Listing 5: User2 applies

```
<message id=”6” from=”user2@chatservice.example” to=”
    chatroom@chatservice.example”>
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-1”>
      <i-like-this xmlns=’urn:example:like’>Very much</i-like-this>
  </apply-to>
</message>
```

3.4 Removing fastenings

Fastenings of a given type (qualified name) can be removed by sending a new (empty) fastening of the same qualified name only (no other attributes and no children) and setting an attribute ’clear’ with value ’true’ on the <apply-to;> element. This logically replaces all the previous fastenings from this sender of this qualified name with an empty list of fastenings.
For example, if user2 wished to remove the fastenings from the first example they would send:

Listing 6: User2 applies

```
<message id=”6” from=”user2@chatservice.example” to=”
    chatroom@chatservice.example”>
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-1” clear=”true”>
```

```
      <i-like-this xmlns='urn:example:like'/>
  </apply-to>
</message>
```

3.5 Interaction with stanza encryption

Where the contents of a fastening are sensitive and must be encrypted (e.g. if a message editing spec. uses fastening, the contents of the edit are sensitive), it is suggested that the plaintext contains an <apply-to> with no content other than the id and an attribute of ”shell” whose value is ”true”, to indicate that the apply-to is a shell with no content. This allows a server to still collate the fastened stanzas, while not allowing it to read the content. The encrypted content MUST then include the full <apply-to> element, which a client MUST use post-decryption and ignore the shell element.

Listing 7: Encrypted fastening

```
<message id=”9” from=”user2@chatservice.example” to=”
   chatroom@chatservice.example”>
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-1” shell=”true”/>
  <!-{}-This is encrypted-{}->
  <apply-to xmlns=”urn:xmpp:fasten:0” id=”origin-id-1”>
      <i-like-this xmlns='urn:example:like'/>
  </apply-to>
  <!-{}-end encryption-{}->
</message>
```

4 Business Rules

In order for a message to be able to have fastenings applied to it it must contain a Unique and Stable Stanza IDs (XEP-0359) [4] stanza id.
A message must only contain a single ”apply-to” element (i.e. a message cannot be fastened to multiple other messages).
Fastenings are not to be chained - an <apply-to> element MUST reply to the original message to which it applies, not to an earlier fastening-containing-message (the 'id' attribute of an <apply-to> element must not be that of a stanza that itself contains an <apply-to>). For example, if this spec is used in the future to wrap multiple subsquent message edits for the same source message, each <apply-to> would contain the original stanza's id, not the id of any subsequent fastening stanza.
Some payloads should not be allowed to be fastened to a particular message by all users - e.g. if used to wrap message edits, it would be expected that only the original author be allowed to edit their message. It's the responsibility of specifications for protocols to be fastened to destribe such ”who is allowed to do this?” applicability rules.

[4] XEP-0359: Unique and Stable Stanza IDs <https://xmpp.org/extensions/xep-0359.html>.

Multiple fastenings of the same type may be present in a single attach-to element, but fastening types may not be mixed in this way.
TODO: (future XEP) special handling by the archive to allow grouping and querying of things fastened to a message (this is going to involve following the externals and potentially wrapping them inside the apply-to when fetching from the archive).

5 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [5].

6 XMPP Registrar Considerations

6.1 Protocol Namespaces

The XMPP Registrar [6] includes 'urn:xmpp:fasten:0' in its registry of protocol namespaces (see <https://xmpp.org/registrar/namespaces.html>).

- urn:xmpp:fasten:0

7 Acknowledgements

The authors would like to thank Marvin Wissfeld, Florian Schmaus, Philipp Hörist, Jonas Schäfer, Ralph Meijer and others who've replied on-list for their feedback.

[5] The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.

[6] The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <https://xmpp.org/registrar/>.