# XEP-0477: Pubsub Targeted Encryption

Jérôme Poisson
mailto:goffi@goffi.org
xmpp:goffi@jabber.fr
https://www.goffi.org

2022-12-20
Version 0.1.0

| Status | Type | Short Name |
| --- | --- | --- |
| Experimental | Standards Track | pte |

Specifies a way to encrypt pubsub items for a restricted set of entities

# Legal

## Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 by the XMPP Standards Foundation (XSF).

## Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the ”Specification”), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

## Warranty

## NOTE WELL: This Specification is provided on an ”AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. ##

## Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

## Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF’s Intellectual Property Rights Policy (a copy of which can be found at <https://xmpp.org/about/xsf/ipr-policy> or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

## Contents


1 Introduction 1

2 Requirements 1

3 Use Cases 1

3.1 Encrypting a Pubsub Item . . . . . 1

4 Business Rules 2

5 Discovering Support 3

6 Security Considerations 3

7 IANA Considerations 3

8 XMPP Registrar Considerations 3

9 XML Schema 4

10 Acknowledgements 4

# 1 Introduction

While it is nowadays possible to encrypt pubsub items with OpenPGP for XMPP Pubsub, this specification is designed for pubsub nodes were all items are end-to-end encrypted, and it is using symmetric encryption with a system of key sharing, meaning that if a key is available, it can decrypt all items encrypted with it.
This is fine for most use cases, however it may be desirable to only encrypt a few items with properties such as Perfect Forward Secrecy. This specification describes a way on how to do that by adapting existing end-to-end encryption algorithms used in instant messaging to pubsub items. This may be used to implement restricted items (a feature known is some other software such as ”aspects” or ”circles”) or for transient nodes.

# 2 Requirements

The design goal of this specification is do adapt simply existing e2e encryption algorithms used for messages to pubsub items.

# 3 Use Cases

## 3.1 Encrypting a Pubsub Item

Juliet holds a public blog using Microblogging Over XMPP (XEP-0277) [1]. However, she wants to publish a new items that should be visible only to some well targeted users. To do so she encrypt the payload in the same way as she encrypt messages with algorithm such as OMEMO Encryption (XEP-0384) [2]. She wrap the encrypted payload in an <encrypted/> element qualified by the 'urn:xmpp:pte:0' namespace which MUST have a 'by' attribute with its own bare jid as value, and which MUST have a 'type' attribute whose value is the namespace of the algorithm used.
She decides to use OMEMO Encryption (XEP-0384) [3] to encrypt her items, her client publish an item like this:

Listing 1: Juliet Publish a Targeted Encrypted Item

```
<iq xmlns=”jabber:client” id=”pte_1” type=”set” from=”juliet@capulet.
    lit/chamber” to=”juliet@capulet.lit”>
  <pubsub xmlns=”http://jabber.org/protocol/pubsub”>
    <items node=”urn:xmpp:microblog:0”>
      <item id=”secret_blog_post” publisher=”juliet@capulet.lit/
          chamber”>
```

[1] XEP-0277: Microblogging over XMPP <https://xmpp.org/extensions/xep-0277.html>.
[2] XEP-0384: OMEMO Encryption <https://xmpp.org/extensions/xep-0384.html>.
[3] XEP-0384: OMEMO Encryption <https://xmpp.org/extensions/xep-0384.html>.

```
        <encrypted xmlns="urn:xmpp:pte:0" by="juliet@capulet.lit" type
           ="urn:xmpp:omemo:2">
          <encrypted xmlns="urn:xmpp:omemo:2">
            <header sid="878841001">
              <keys jid="juliet@capulet.lit">
                <key rid="673880319">ChDRqSBLTR+
                    RtRIH8io7kf22EmgIARACGiCasIYfB6Zfe5SNyT8twIa+
                    mEYA8h7uEQIjQ64dJx4vXiJAZSpXPRW+
                    sVVSC7gc4lDEiTA4DT7AIh/
                    woa82PFjgFdL0A8HTyBe7yh3UWThZGuTp5A3zmjXH7pAwKX85oxQ8XA
                    ==</key>
              </keys>
            </header>
            <payload>
                DWmAVvrKlPPh23OmvmIrJmQXj5hVtgAnY8IOGZNqJc59T93hzsTen7tw7Kea5KfM3btfS2
                /GJM2GAT55exZSRU0Px8/
                E8j0XMtHCuZ4j3z0EBk1NZin0suQv8rVy1liWACPNuVrnU7h8LpdWmUggYztqL9l1yoxzE
                /
                LmdYspVUPzPpQt7OmAKAndFWXTsAV5wmbtVsr15TzxI4NVDZyp7G70TYyTHlhG2gAq7StV
                /ZG8pbe/GXUoPg4q9ZfuDiOYBHUugUxNsVFactRp6UocaQT/
                RogrqKY3o6NlTvnqVYpMJsi72cp8uQWTPtqwBpxyhAY0jKp1D+
                y7m2wzbeD2SZCw5+FryXulQhCKJ0dLI00PJr4dELWdu+
                uQLdyHl5FxG4D8mLQVOnY/
                TMa0vXUxsMAQI8g8LEHdJIhKU4GyVt125WhrbMrbcBu8iKCYmiz820siZeD8i5iZa1eQ69
                +0
                pHcyzpNC8408B7LhkgwxOEopExdOfv1NFwamsN5zXhCqj386oGRl9Ry0Gw
                +QSv9jlW4FB0rM8r+GF5gB66p0nYU/U5W8efXgNI/
                W1yAdUxgXc9FiQMmzIauTmR4m5WUxPjBggVYz1q3TkeZHQJpWy47EWZPnM9leWKNqC
                </payload>
            </encrypted>
          </encrypted>
        </item>
      </items>
    </pubsub>
</iq>
```

## 4 Business Rules

The properties of the encryption algorithm applies. For instance in the case of OMEMO Encryption (XEP-0384) [4], there Perfect Forward Secrecy, meaning that once an item has been decrypted once by a targeted entity, it can't be decrypted anymore. Client shoud then handle pubsub caching of the decrypted item when necessary.

[4] XEP-0384: OMEMO Encryption <https://xmpp.org/extensions/xep-0384.html>.

## 5 Discovering Support

If a client supports the protocol specified in this XEP, it MUST advertise it by including the ”urn:xmpp:pte:0” discovery feature in response to a Service Discovery (XEP-0030) [5] information request, Then the supported encryption algorithms are announced as explained in their respective XEPs.

Listing 2: Service Discovery information request

```
<iq type='get'
    from='juliet@example.org/balcony'
    to='romeo@example.org/orchard'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'/>
</iq>
```

Listing 3: Service Discovery information response

```
<iq type='result'
    from='romeo@example.org/orchard'
    to='juliet@example.org/balcony'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    ...
    <feature var='urn:xmpp:pte:0'/>
    <feature var='urn:xmpp:omemo:2'/>
    ...
  </query>
</iq>
```

## 6 Security Considerations

Security Considerations of used encryption specifications apply.

## 7 IANA Considerations

TODO

## 8 XMPP Registrar Considerations

TODO

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

## 9 XML Schema

TODO

## 10 Acknowledgements

Thanks to NLNet foundation/NGI0 Discovery for funding.