Network Working Group
Internet-Draft
Expires: August 4, 2003

P. Saint-Andre
Jabber Software Foundation
J. Hildebrand
Jabber, Inc.
February 03, 2003

End-to-End Object Encryption in XMPP
draft-ietf-xmpp-e2e-00

Status of this Memo

Copyright Notice

Abstract

This document describes an end-to-end object encryption method for use in the eXtensible Messaging and Presence Protocol (XMPP).

Table of Contents

1. Introduction

This document describes an end-to-end encryption method for use in the eXtensible Messaging and Presence Protocol (XMPP) as defined in XMPP Core [1] and XMPP IM [2]. Object encryption enables a sender to encrypt a message sent to a specific recipient and assists the XMPP specifications in meeting the requirements of RFC 2779 [4]. Object encryption is accomplished by sending the encrypted form of the message body along with a unique message ID to help prevent replay attacks. The public key used for message encryption SHOULD match the KeyID sent when signaling support for this protocol via presence broadcast.

All operations described herein may be completed using standard OpenPGP [3] software. All program output is US-ASCII armored output with the headers removed, which allows for straightforward encapsulation of the program output directly as XML CDATA. It is assumed that keys may be exchanged using OpenPGP key servers; for example, the key of another user may be retrieved automatically when an appropriate presence stanza is received from that user.

1.1 Terminology

This document inherits the terminology defined in XMPP Core [1].

The capitalized 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 RFC 2119 [5].

1.2 Discussion Venue

The authors welcome discussion and comments related to the topics presented in this document. The preferred forum is the <xmppwg@jabber.org> mailing list, for which archives and subscription information are available at <http://www.jabber.org/cgi-bin/mailman/listinfo/xmppwg/>.

1.3 Intellectual Property Notice

This document is in full compliance with all provisions of Section 10 of RFC 2026. Parts of this specification use the term "jabber" for identifying namespaces and other protocol syntax. Jabber[tm] is a registered trademark of Jabber, Inc. Jabber, Inc. grants permission to the IETF for use of the Jabber trademark in association with this specification and its successors, if any.

2. Encrypting Messages

The encrypted payload contains what would be the main <body> element if the message were not encrypted, along with a message ID to help prevent replay attacks; both pieces of information are wrapped in a <payload/> element scoped by the 'http://jabber.org/protocol/e2e#payload' namespace, as shown in the following example:

```
<payload xmlns='http://jabber.org/protocol/e2e#payload'>
  <id>someID</id>
  <body>Wherefore art thou?</body>
</payload>
```

The encrypted payload MUST include an <id/> element. The CDATA of the <id/> element SHOULD be constructed according to the following algorithm: (1) concatenate the sender's full JID (user@host/resource) with the recipient's full JID; (2) concatenate these JID strings with a full ISO-8601 timestamp including year, month, day, hours, minutes, seconds, and UTC offset if appropriate in the following format: yyyy-mm-dd-Thh:mm:ss-hh:mm; (3) hash the resulting string according to the SHA1 algorithm; (4) convert the hexidecimal SHA1 output to all lowercase.

The full <payload/> element (including all XML tag names and angle brackets) MUST then be encrypted according to the OpenPGP algorithm using the sender's KeyID. The armored output MUST be US-ASCII and have the headers removed. The resulting cipher text MUST then be provided as the CDATA of an <x/> element scoped by the 'http://jabber.org/protocol/e2e' namespace.

Finally, the message stanza SHOULD contain an unencrypted <body/> child element whose CDATA informs the recipient that the actual message body is encrypted.

The format of the full message stanza that results from the foregoing procedure is shown in the following example.

```
<message
    from='juliet@capulet.com/balcony'
    to='romeo@montague.net/orchard'>
  <body>Encrypted is this message.</body>
  <x xmlns='http://jabber.org/protocol/e2e'>
hQEOA+fczQLixGb6EAP/Uv0JNo1x/h9d6ia75foKB1sViwAeXnrAwUDuxFhTBdt3
HDOeF61b/sqaHBi4B4L50xn4W+dZd0sxgf4QNoWucI6WfqcV5BT3K62iTGLVJ7Lc
RoXTylekNsDiNsMVMJBHoYqeoRmTuMt3uuljBHHnXVya7XGMmyxbM/QtdxuykssD
/jsvER1EyIfYSWT+G/djvymd9FfgTwLrgyBjC1S0GfQ6oEjmEz5FK+BpwfRDzxjD
eR08Q6m7Y8C84OC4Dq4UCSCcdzhhKHH0pACizjeG/2N+DlEwDkwK3b/2ED8fFPE1
tCUIl6Z8uvAw5Q6OBeFabgbjdi3QjqY32fV5tOtUkkvk0sAEAcRBF9HqEHNDMEb/
bGza03mV58dlEOjhZEu2rCffR4mqYSDoF8hNb/XuOssDuIvp342ILfAPjyx/AE1/
ffdN0tSWt3kEZzDzeJfFOBzv2n8OPNUKrRAoinnRr9vdFH5KlIQbTFte0Fk/r7YA
7PghNWtPZJ/mXQPoCylaK86wGc/KHld8Y+RopWeZSoicpIqGBrpuwdl/o/OtEm0b
VnDh3dJpz89aJj2RAAiTaKLotLg/AkmwfQGLZnmv416jxm6zy1p1rQ==
=1rou
  </x>
</message>
```

The decoded payload is:

```
<payload xmlns='http://jabber.org/protocol/e2e#payload'>
  <id>e0ffe42b28561960c6b12b944a092794b9683a38</id>
  <body>O Romeo, Romeo! Wherefore art thou Romeo?</body>
</payload>
```

3. Signaling Support via Presence

In order to signal support for this method oof encrypting message bodies, an entity MUST broadcast its KeyID in all outgoing presence stanzas, contained in an <x/> element scoped by the 'http://jabber.org/protocol/e2e' namespace.

```
<presence
    from='juliet@capulet.com/balcony'
    to='romeo@montague.net/orchard'>
  <show>away</show>
  <status>be right back</status>
  <x xmlns='http://jabber.org/protocol/e2e'>88CA1D46</x>
</presence>
```

4. Security Considerations

Replay attacks are made more difficult using this method because of the inclusion of a unique ID in the encrypted object. Key exchange may rely on the web of trust model used on the OpenPGP keys network. There is no method to check a fingerprint or ownership of a key other than checking the user IDs on a key.

References

[1] Saint-Andre, P. and J. Miller, "XMPP Core (draft-ietf-xmpp-core-02, work in progress)", February 2003.

[2] Saint-Andre, P. and J. Miller, "XMPP Instant Messaging (draft-ietf-xmpp-im-02, work in progress)", February 2003.

[3] Elkins, M., Del Torto, D., Levien, R. and T. Roessler, "MIME Security with OpenPGP", RFC 3156, August 2001.

[4] Day, M., Aggarwal, S., Mohr, G. and J. Vincent, "A Model for Presence and Instant Messaging", RFC 2779, February 2000, <http://www.ietf.org/rfc/rfc2779.txt>.

[5] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Peter Saint-Andre
Jabber Software Foundation

EMail: stpeter@jabber.org
URI: http://www.jabber.org/people/stpeter.php

Joe Hildebrand
Jabber, Inc.

EMail: jhildebrand@jabber.com
URI: http://www.jabber.org/people/hildjj.php

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.