XMPP	P. Saint-Andre
Internet-Draft	Cisco
Intended status: Informational	August 28, 2009
Expires: March 1, 2010	

TOC

Requirements for End-to-End Encryption in the Extensible Messaging and Presence Protocol (XMPP)
draft-ietf-xmpp-e2e-requirements-00

Status of this Memo

Copyright Notice

Abstract

This document describes requirements for end-to-end encryption in the Extensible Messaging and Presence Protocol (XMPP).

Table of Contents

---

1. Introduction

TOC

End-to-end or "e2e" encryption of traffic sent over the Extensible Messaging and Presence Protocol (XMPP) is a desirable goal. Since 1999, the Jabber/XMPP developer community has experimented with several such technologies, including OpenPGP [XMPP-PGP] (Muldowney, T., “Current Jabber OpenPGP Usage,” November 2006.), S/MIME [XMPP-SMIME] (Saint-Andre, P., “End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP),” October 2004.), and encrypted sessions [ESessions] (Paterson, I., Saint-Andre, P., and D. Smith, “Encrypted Session Negotiation,” May 2007.). More recently, the community has explored the possibility of using Transport Layer Security [TLS] (Dierks, T. and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.2,” August 2008.) as the base technology for e2e encryption. In order to provide a foundation for deciding on a sustainable approach to e2e encryption, this document specifies a set of requirements that the ideal technology would meet. The preferred venue for discussion of this document is the xmpp@ietf.org mailing list; visit https://www.ietf.org/mailman/listinfo/xmpp for further information.
Much of the text in this document has been copied from [XEP-0210] (Paterson, I., “Requirements for Encrypted Sessions,” May 2007.).

---

2. Scope

TOC

There are several different kinds of communications between XMPP entitites:

1. One-to-one communication sessions between two entities, where each entity is online and available during the life of the session so that all of the communications occur in real time.

2. One-to-one messages that are not transferred in real time but that instead are stored when sent and then forwarded when the recipient is next online; these are usually called "offline messages" as described in [OFFLINE] (Saint-Andre, P., "Best Practices for Handling Offline Messages," January 2006.).

3. One-to-many information broadcast, such as undirected presence stanzas sent from one user to many contacts as described in [XMPP-IM] (Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence," June 2009.) and data syndication as described in [PubSub] (Millard, P., Saint-Andre, P., and R. Meijer, "Publish-Subscribe," September 2008.).

4. Many-to-many communication sessions among more than two entities, such as a text conference in a chatroom as described in [MUC] (Saint-Andre, P., "Multi-User Chat," July 2008.).

Ideally, any technology for end-to-end encryption in XMPP could be extended to cover all the scenarios above. However, both one-to-many broadcast and many-to-many sessions are deemed out-of-scope for this document, and this document puts more weight on one-to-one communication sessions (the typical scenario for XMPP) than on offline messages.

---

3. Threat Analysis

TOC

XMPP technologies are typically deployed using a client-server architecture. As a result, XMPP endpoints (often but not always controlled by human users) need to communicate through one or more servers. For example, the user juliet@capulet.lit connects to the capulet.lit server and the user romeo@montague.lit connects to the montague.lit server, but in order for Juliet to send a message to Romeo the message will be routed over her client-to-server connection with capulet.lit, over a server-to-server connection between capulet.lit and montague.lit, and over Romeo's client-to-server connection with montague.lit. Although [XMPP-CORE] (Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core," June 2009.) requires support for Transport Layer Security [TLS] (Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," August 2008.) to make it possible to encrypt all of these connections, when XMPP is deployed any of these connections might be unencrypted. Furthermore, even if the server-to-server connection is encrypted and both of the client-to-server connections are encrypted, the message would still be in the clear while processed by both the capulet.lit and montague.lit servers.

In this specification we primarily address communications security ("commsec") between two parties, especially confidentiality, data integrity, and peer entity authentication. Communications security can be subject to a variety of attacks, which [RFC3552] (Rescorla, E. and B. Korver, “Guidelines for Writing RFC Text on Security Considerations,” July 2003.) divides into passive and active categories. In a passive attack, information is leaked (e.g., a passive attacker could read all of the messages that Juliet sends to Romeo). In an active attack, the attacker can add, modify, or delete messages between the parties, thus disrupting communications.

Traditionally, it seems that XMPP users have been concerned more about passive attacks (such as eavesdropping) than about active attacks (such as man-in-the-middle), perhaps because they have thought that their communications are "just chat", because they have had no expectation that endpoints could be authenticated, or because they have believed that hijacked communications would be detected socially (e.g., because the other party did not have an authentic "voice" in a text conversation). However, both forms of attack are of concern in this protocol.

In particular, we consider the following types of attacks and attackers:

*One type of passive attack might involve monitoring all the conversations of a given party. To help prevent this, it is important for the party to ensure that its connection with its server is protected using TLS. However, in this case the eavesdropper could monitor outbound traffic from the party's server, either to other connected clients or to other servers, since that traffic might be unencrypted. In addition, the eavesdropper could attack the party's server so that it gains access to all traffic within the server, or masquerade as the party's server so that the party is fooled into connecting to the attacker rather than directly to the party's server.

*Another type of passive attack might involve monitoring of a single conversation between two particular parties. In this case the eavesdropper could monitor communications over the server-to-server connection between the parties' servers, or over the client-to-server connection between either party and that party's server.

*One type of active attack would involve modification of the XML stanzas used to advertise support for the protocol "building blocks" that make it possible to negotiate a secure session; as a result, other parties would be led to believe that the party does not have the ability to negotate a secure session and therefore would not attempt such a negotiation.

*Another type of active attack would involve modification or outright deletion of the XML stanzas used to negotiate a secure session (such as those described in this document), with the result that the parties would think the negotiation has failed for legitimate reasons such as incompatibilities between the parties' clients.

*A more sophisticated active attack would involve a cryptanalytic attack on the keying material or other credentials used to establish trust between the parties, such as an ephemeral password exchanged during an initial certificate exchange if Secure Remote Password [TLS-SRP] (Taylor, D., Wu, T., Mavrogiannopoulos, N., and T. Perrin, "Using the Secure Remote Password (SRP) Protocol for TLS Authentication," November 2007.) is used.

Other attacks are possible, and the foregoing list is best considered incomplete at this time.

---

4. Security Requirements

TOC

This document stipulates the following security requirements for end-to-end encryption of XMPP communications:

Confidentiality: The one-to-one XML stanzas exchanged between two entities (conventionally, "Alice" and "Bob") must not be understandable to any other entity that might intercept the communications. The encrypted stanzas should be understood by an intermediate server only to the extent required to route them.

Integrity: Alice and Bob must be sure that no other entity can change the content of the XML stanzas they exchange, or remove or insert stanzas undetected.

Replay Protection: Alice or Bob must be able to identify and reject any communications that are copies of their previous communications resent by another entity.

Perfect Forward Secrecy: The encrypted communication should not be revealed even if long-lived keys are compromised in the future (e.g., Steve steals Bob's computer). For long-lived sessions it must be possible to periodically change the decryption keys.

PKI Independence: The protocol must not force the use of any public key infrastructure (PKI), certification authority, web of trust, or any other trust model that is external to the trust established between Alice and Bob. However, if external

authentication or trust models are available then Alice and Bob should be able to use such trust models to enhance any trust that exists between them.

Authentication: Each party to a conversation must know that the other party is who they want to communicate with (Alice must be able to know that Bob really is Bob, and vice versa). Note: Authentication can be as simple as Alice confirming that Bob is the same Bob that she communicated with yesterday or that she talked to on the telephone. The reliable association between an entity and its public keys is "identification" and therefore beyond the scope of this document.

Identity Protection: No other entity should be able to identify Alice or Bob. The JabberIDs they use to route their stanzas are unavoidably vulnerable to interception. Therefore, even if Alice and Bob protect their identities by using different JabberIDs for each session, it must be possible for their user agents to authenticate them transparently, without any other entity identifying them via an active ("man-in-the-middle") attack, or even linking them to their previous sessions. If that is not possible because Alice and Bob choose to authenticate using public keys instead of retained shared secrets, then the public keys must not be revealed to other entities using a passive attack. Bob should also be able to choose between protecting either his public key or Alice's public key from disclosure through an active attack.

Robustness: The protocol should provide more than one difficult challenge that has to be overcome before an attack can succeed (for example, by generating encryption keys using as many shared secrets as possible, such as retained secrets or optional passwords).

Upgradability: The protocol must be upgradable so that, if a vulnerability is discovered, a new version can fix it. Alice must tell Bob which versions of the protocol she is prepared to support.

---

5. Application Requirements

TOC

In addition to the foregoing security profile, this document also stipulates the following application-specific requirements:

Generality: The solution must be generally applicable to the full content of any XML stanza type (<message/>, <presence/>, and <iq/

>) sent between two entities. It is deemed acceptable if the solution does not apply to many-to-many stanzas (e.g., groupchat messages sent within the context of multi-user chat) or one-to-many stanzas (e.g., presence "broadcasts" and publish-subscribe notifications); end-to-end encryption of such stanzas might require separate solutions.

Implementability: The only good security technology is an implemented security technology. The solution should be one that XMPP client developers can implement in a relatively straightforward and interoperable fashion. Ideally the solution would reuse existing technologies so that client developers can also reuse existing libraries, as they already do for security features such as Transport Layer Security [TLS] (Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," August 2008.) and the Simple Authentication and Security Layer [SASL] (Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)," June 2006.).

Usability: The requirement of usability takes implementability one step further by stipulating that the solution should be one that organizations can deploy and humans can use with the ease-of-use of, say, "https:" URLs. Experience has shown that solutions requiring a full public key infrastructure do not get widely deployed and that solutions requiring any user action are not widely used. If, however, Alice and/or Bob are prepared to verify the integrity of their copies of each other's keys (thus enabling them to discover targeted active attacks or even the mass surveilance of a population), then the actions necessary for them to achieve that should be minimal (requiring no more effort than a one-time out-of-band verification of a string of up to 8 alphanumeric characters).

Efficiency: Cryptographic operations are highly CPU intensive, particularly public key and Diffie-Hellman operations. Cryptographic data structures can be relatively large, especially public keys and certificates. Network round trips can introduce unacceptable delays, especially over high-latency wireless connections. The solution must perform efficiently even when CPU and network bandwidth are constrained. The number of stanzas required for negotiation of encrypted communication should be minimized.

Flexibility: The solution must be compatible with a variety of existing and future cryptographic algorithms and identity certification schemes, including [X509] (Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," May 2008.) and [OpenPGP] (Callas,

J., Donnerhacke, L., Finney, H., Shaw, D., and R. Thayer, "OpenPGP Message Format," November 2007.). The protocol must also be able to evolve to correct the weaknesses that are inevitably discovered once any cryptographic protocol is in widespread use.

Offline messages: It should be possible to encrypt one-to-one communications that are stored for later delivery (so-called "offline messages") and still benefit from Perfect Forward Secrecy (with a slightly longer period of vulnerability than if both parties were online simultaneously). However, any vulnerabilities introduced into the solution in order to enable such offline communications must not make real-time communications more vulnerable.

---

6. Security Considerations

TOC

Security issues are discussed throughout this document.

---

7. IANA Considerations

TOC

This document has no actions for the IANA.

---

8. Informative References

TOC

[ESessions]	Paterson, I., Saint-Andre, P., and D. Smith, "Encrypted Session Negotiation," XSF XEP 0116, May 2007.
[MUC]	Saint-Andre, P., "Multi-User Chat," XSF XEP 0045, July 2008.
[OFFLINE]	Saint-Andre, P., "Best Practices for Handling Offline Messages," XSF XEP 0160, January 2006.
[OpenPGP]	Callas, J., Donnerhacke, L., Finney, H., Shaw, D., and R. Thayer, "OpenPGP Message Format," RFC 4880, November 2007 (TXT).
[PubSub]	Millard, P., Saint-Andre, P., and R. Meijer, "Publish-Subscribe," XSF XEP 0060, September 2008.
[RFC3552]	Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations," BCP 72, RFC 3552, July 2003 (TXT).
[SASL]	Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)," RFC 4422, June 2006 (TXT).
[TLS]	

	Dierks, T. and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.2,” RFC 5246, August 2008 (TXT).
[TLS-SRP]	Taylor, D., Wu, T., Mavrogiannopoulos, N., and T. Perrin, “Using the Secure Remote Password (SRP) Protocol for TLS Authentication,” RFC 5054, November 2007 (TXT).
[X509]	Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile,” RFC 5280, May 2008 (TXT).
[XEP-0210]	Paterson, I., “Requirements for Encrypted Sessions,” XSF XEP 0210, May 2007.
[XMPP-CORE]	Saint-Andre, P., “Extensible Messaging and Presence Protocol (XMPP): Core,” draft-ietf-xmpp-3920bis-00 (work in progress), June 2009 (TXT).
[XMPP-IM]	Saint-Andre, P., “Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence,” draft-ietf-xmpp-3921bis-00 (work in progress), June 2009 (TXT).
[XMPP-PGP]	Muldowney, T., “Current Jabber OpenPGP Usage,” XSF XEP 0027, November 2006.
[XMPP-SMIME]	Saint-Andre, P., “End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP),” RFC 3923, October 2004 (TXT, HTML, XML).

Author's Address

TOC

	Peter Saint-Andre
	Cisco
Email:	psaintan@cisco.com