Network Working Group
Internet-Draft
Updates: 6120 (if approved)
Intended status: Standards Track
Expires: April 22, 2014

P. Saint-Andre
Cisco Systems, Inc.

October 19, 2013

Use of Transport Layer Security (TLS) in the Extensible Messaging and Presence Protocol (XMPP)
draft-saintandre-xmpp-tls-02

Abstract

This document provides recommendations for the use of Transport Layer Security (TLS) in the Extensible Messaging and Presence Protocol (XMPP). This document updates RFC 6120.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 22, 2014.

Table of Contents

1. Introduction 3
2. Terminology 3
3. Discussion Venue 3
4. Recommendations 3
4.1. Support for TLS 3
4.2. Protocol Versions 4
4.3. Ciphersuites 4
4.4. Public Key Length 6
4.5. Certificate Validation 6
4.6. Unauthenticated Connections 6
4.7. Server Name Indication 7
4.8. Session Resumption 7
4.9. Compression 7
4.10. Human Factors 7
5. Implementation Notes 8
6. IANA Considerations 8
7. Security Considerations 8
8. References 8
8.1. Normative References 8
8.2. Informative References 9
Appendix A. Acknowledgements 10
Author's Address 10

1. Introduction

The Extensible Messaging and Presence Protocol (XMPP) [RFC6120] (along with its precursor, the so-called "Jabber protocol") has used Transport Layer Security (TLS) [RFC5246] (along with its precursor, Secure Sockets Layer or SSL) since 1999. Both [RFC6120] and its predecessor [RFC3920] provided recommendations regarding the use of TLS in XMPP. Given the evolving threat model on the Internet today (see, for example, [I-D.trammell-perpass-ppa]), it is necessary to provide stronger recommendations (see also [I-D.sheffer-tls-bcp]). This document updates [RFC6120].

2. Terminology

Various security-related terms are to be understood in the sense defined in [RFC4949].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Discussion Venue

The discussion venue for this document is the mailing list of the XMPP Working Group, for which archives and subscription information can be found at <https://www.ietf.org/mailman/listinfo/xmpp>.

4. Recommendations

4.1. Support for TLS

Support for TLS (specifically, the XMPP profile of STARTTLS) is mandatory for XMPP implementations. If the server to which an XMPP client or peer server connects does not offer a stream feature of <starttls xmlns='urn:ietf:params:xml:ns:xmpp-tls'/> as described in [RFC6120], the initiating entity MUST NOT proceed with the stream negotiation and MUST instead abort the connection attempt. Although XMPP servers SHOULD include the <required/> child element to indicate that negotiation of TLS is mandatory, clients and peer servers MUST NOT depend on receiving the <required/> flag.

4.2. Protocol Versions

It is important both to stop using old, insecure versions of SSL/TLS and to start using modern, more secure versions. Therefore:

- XMPP implementations MUST NOT negotiate SSL version 2.

  Rationale: SSLv2 has serious security vulnerabilities [RFC6176].

- XMPP implementations MUST NOT negotiate SSL version 3.

  Rationale: SSLv3 [RFC6101] was an improvement over SSLv2, but did not support strong ciphersuites.

- XMPP implementations SHOULD NOT negotiate TLS version 1.0 [RFC2246].

  Rationale: TLS 1.0 (published in 1999) includes a way to downgrade the connection to SSLv3 and does not support more modern, strong ciphersuites.

- XMPP implementations MAY negotiate TLS version 1.1 [RFC4346].

  Rationale: TLS 1.1 (published in 2006) prevents downgrade attacks to SSL, but does not support certain stronger ciphersuites.

- XMPP implementations MUST support, and prefer to negotiate, TLS version 1.2 [RFC5246].

  Rationale: Several stronger ciphersuites are available only with TLS 1.2 (published in 2008).

As of the date of this writing, the latest version of TLS is 1.2. When TLS is updated to a newer version, this document will be updated to recommend support for the latest version. If this document is not updated in a timely manner, it can be assumed that support for the latest version of TLS is recommended.

4.3. Ciphersuites

It is important both to stop using old, insecure ciphersuites and to start using modern, more secure ciphersuites. Therefore:

- XMPP implementations MUST NOT negotiate the NULL ciphersuites.

  Rationale: The NULL ciphersuites offer no encryption whatsoever and thus are completely insecure.

- XMPP implementations MUST NOT negotiate RC4 ciphersuites

  Rationale: The RC4 stream cipher has a variety of cryptographic weaknesses, documented in [I-D.popov-tls-prohibiting-rc4].

- XMPP implementations MUST NOT negotiate ciphersuites that use so-called "export-level" encryption (including algorithms with 40 bits or 56 bits of security).

  Rationale: These ciphersuites are deliberately "dumbed down" and are very easy to break.

- XMPP implementations MUST NOT negotiate ciphersuites that use algorithms that offer less than 128 bits of security (even if they advertise more bits, such as the 168-bit 3DES ciphersuites).

  Rationale: Although these ciphersuites are not actively subject to breakage, their useful life is short enough that stronger ciphersuites are desirable.

- XMPP implementations SHOULD prefer ciphersuites that use algorithms with at least 256 bits of security.

  Rationale: The useful life of such ciphersuites is probably at least 3-5 years.

- XMPP implementations MUST support, and SHOULD prefer to negotiate, ciphersuites that offer authentication, such as the "AES-GCM" family.

  Rationale: Authenticated connections are better than unauthenticated connections (although, as explained under Section 4.6, unauthenticated connections are better than nothing).

- XMPP implementations MUST support, and SHOULD prefer to negotiate, ciphersuites that offer forward secrecy, such as those in the "EDH", "DHE", and "ECDHE" families.

  Rationale: Forward secrecy (sometimes called "perfect forward secrecy") prevents the recovery of information that was encrypted with older keys, thus limiting the amount of time during which attack can be successful.

Given the foregoing considerations, implementation of the following ciphersuites is RECOMMENDED:

- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

Unfortunately, those ciphersuites are supported only in TLS 1.2. A future version of this document might recommend ciphersuites for earlier versions of TLS.

4.4. Public Key Length

Because Diffie-Hellman keys of 1024 bits are estimated to be roughly equivalent to 80-bit symmetric keys, it is better to use longer keys for the "DH" family of ciphersuites. Unfortunately, some existing software cannot handle (or cannot easily handle) key lengths greater than 1024 bits. The most common workaround for these systems is to prefer the "ECDHE" family of ciphersuites instead of the "DH" family, then use longer keys. Key lengths of at least 2048 bits are RECOMMENDED, since they are estimated to be roughly equivalent to 112-bit symmetric keys and might be sufficient for at least the next 10 years.

Note: The foregoing recommendations are preliminary and will likely be corrected and enhanced in a future version of this document.

4.5. Certificate Validation

Both the core XMPP specification [RFC6120] and the "CertID" specification [RFC6125] provide recommendations and requirements for certificate checking. This document does not supersede those specifications.

4.6. Unauthenticated Connections

The core XMPP specification [RFC6120] states a preference for the use of TLS for encryption along with SASL [RFC4422] for authentication. In general, it is preferable for a connection to be authenticated, including proper identity checking as defined by the "CertID" specification [RFC6125]. However, given the pervasiveness of passive eavesdropping, even an unauthenticated connection might be better than an unencrypted connection (this is similar to the "better than nothing security" approach for IPsec [RFC5386]). In particular, given current deployment challenges for authenticated connections between XMPP servers [I-D.ietf-xmpp-dna], it might be reasonable for XMPP server implementations to accept unauthenticated connections when the Server Dialback protocol [XEP-0220] is used for weak identity verification; this will at least enable encryption of server-to-server connections. Unauthenticated connections include

connections negotiated using anonymous Diffie-Hellman algorithms or using self-signed certificates, among other scenarios.

4.7. Server Name Indication

Although there is no harm in supporting the TLS Server Name Indication (SNI) extension [RFC6066], this is not necessary since the same function is served in XMPP by the 'to' address of the initial stream header as explained in Section 4.7.2 of [RFC6120].

4.8. Session Resumption

If TLS session resumption is used (e.g., in concert with the XMPP Stream Management extension [XEP-0198]), care ought to be taken to do so safely. In particular, the resumption information (either session IDs [RFC5246] or session tickets [RFC5077]) needs to be authenticated and encrypted to prevent modification or eavesdropping by an attacker. Use of session IDs [RFC5246] is RECOMMENDED instead of session tickets [RFC5077], since session tickets require use of a relatively small key size and a relatively weak ciphersuite (AES_128_CBC_SHA256) that does not support forward secrecy.

4.9. Compression

XMPP is not generally subject to attacks based on TLS-layer compression (e.g., the "CRIME" attack), since it is not typically used to communicate static strings of the kind communicated over HTTP, such as "cookies" [RFC6265]. However, because XMPP also supports an application-layer compression technology [XEP-0138], implementers might wish to prefer XMPP compression over TLS compression.

4.10. Human Factors

It is RECOMMENDED that XMPP clients provide ways for end users (and that XMPP servers provide ways for administators) to complete the following tasks:

- Determine if a client-to-server or server-to-server connection is encrypted and authenticated.
- Determine the version of TLS used for a client-to-server or server-to-server connection.
- Inspect the certificate offered by an XMPP server.
- Determine the ciphersuite used to encrypt a connection.
- Be warned if the certificate changes for a given server.

5. Implementation Notes

Some governments enforce legislation prohibiting the export of strong cryptographic technologies. Nothing in this document ought to be taken as advice to violate such prohibitions.

6. IANA Considerations

This document requests no actions of the IANA.

7. Security Considerations

This entire document discusses security.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.

[RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, January 2008.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, March 2011.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, March 2011.

[RFC6176] Turner, S. and T. Polk, "Prohibiting Secure Sockets Layer (SSL) Version 2.0", RFC 6176, March 2011.

8.2. Informative References

[I-D.ietf-xmpp-dna]
Saint-Andre, P. and M. Miller, "Domain Name Associations (DNA) in the Extensible Messaging and Presence Protocol (XMPP)", draft-ietf-xmpp-dna-03 (work in progress), September 2013.

[I-D.popov-tls-prohibiting-rc4]
Popov, A., "Prohibiting RC4 Cipher Suites", draft-popov-tls-prohibiting-rc4-01 (work in progress), October 2013.

[I-D.sheffer-tls-bcp]
Sheffer, Y., "Recommendations for Secure Use of TLS and DTLS", draft-sheffer-tls-bcp-01 (work in progress), September 2013.

[I-D.trammell-perpass-ppa]
Trammell, B., "The Perfect Passive Adversary: A Threat Model for the Evaluation of Protocols under Pervasive Surveillance", draft-trammell-perpass-ppa-00 (work in progress), September 2013.

[RFC2246] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, January 1999.

[RFC3920] Saint-Andre, P., Ed., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 3920, October 2004.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC4422] Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)", RFC 4422, June 2006.

[RFC5386] Williams, N. and M. Richardson, "Better-Than-Nothing Security: An Unauthenticated Mode of IPsec", RFC 5386, November 2008.

[RFC6066] Eastlake, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, January 2011.

[RFC6101] Freier, A., Karlton, P., and P. Kocher, "The Secure Sockets Layer (SSL) Protocol Version 3.0", RFC 6101, August 2011.

[RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265,

April 2011.

[XEP-0138]
Hildebrand, J. and P. Saint-Andre, "Stream Compression", XSF XEP 0138, May 2009.

[XEP-0198]
Karneges, J., Saint-Andre, P., Hildebrand, J., Forno, F., Cridland, D., and M. Wild, "Stream Management", XSF XEP 0198, June 2011.

[XEP-0220]
Miller, J., Saint-Andre, P., and P. Hancke, "Server Dialback", XSF XEP 0220, September 2013.

Appendix A. Acknowledgements

Thanks to the following individuals for their input: Thijs Alkemade, Dave Cridland, Philipp Hancke, Olle Johansson, Steve Kille, Tobias Markmann, Matt Miller, and Rene Treffer.

Author's Address

Peter Saint-Andre
Cisco Systems, Inc.
1899 Wynkoop Street, Suite 600
Denver, CO 80202
USA

Phone: +1-303-308-3282
Email: psaintan@cisco.com