Internet Engineering Task Force (IETF)
Request for Comments: 7568
Updates: 5246
Category: Standards Track
ISSN: 2070-1721

R. Barnes
M. Thomson
Mozilla
A. Pironti
INRIA
A. Langley
Google

June 2015

Deprecating Secure Sockets Layer Version 3.0

Abstract

The Secure Sockets Layer version 3.0 (SSLv3), as specified in RFC 6101, is not sufficiently secure. This document requires that SSLv3 not be used. The replacement versions, in particular, Transport Layer Security (TLS) 1.2 (RFC 5246), are considerably more secure and capable protocols.

This document updates the backward compatibility section of RFC 5246 and its predecessors to prohibit fallback to SSLv3.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7568.

Table of Contents

1. Introduction

Since it was released in 1996, the SSLv3 protocol [RFC6101] has been subject to a long series of attacks, both on its key exchange mechanism and on the encryption schemes it supports. Despite being replaced by TLS 1.0 [RFC2246] in 1999, and subsequently TLS 1.1 in 2002 [RFC4346] and 1.2 in 2006 [RFC5246], availability of these replacement versions has not been universal. As a result, many implementations of TLS have permitted the negotiation of SSLv3.

The predecessor of SSLv3, SSL version 2, is no longer considered sufficiently secure [RFC6176]. SSLv3 now follows.

2. Terminology

The 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 [RFC2119].

3. Do Not Use SSL Version 3.0

SSLv3 MUST NOT be used. Negotiation of SSLv3 from any version of TLS MUST NOT be permitted.

Any version of TLS is more secure than SSLv3, though the highest version available is preferable.

Pragmatically, clients MUST NOT send a ClientHello with ClientHello.client_version set to {03,00}. Similarly, servers MUST NOT send a ServerHello with ServerHello.server_version set to {03,00}. Any party receiving a Hello message with the protocol version set to {03,00} MUST respond with a "protocol_version" alert message and close the connection.

Historically, TLS specifications were not clear on what the record layer version number (TLSPlaintext.version) could contain when sending ClientHello. Appendix E of [RFC5246] notes that TLSPlaintext.version could be selected to maximize interoperability, though no definitive value is identified as ideal. That guidance is still applicable; therefore, TLS servers MUST accept any value {03,XX} (including {03,00}) as the record layer version number for ClientHello, but they MUST NOT negotiate SSLv3.

4. SSLv3 Is Comprehensively Broken

4.1. Record Layer

The non-deterministic padding used in the Cipher Block Chaining (CBC) construction of SSLv3 trivially permits the recovery of plaintext [POODLE]. More generally, the CBC modes of SSLv3 use a flawed MAC-then-encrypt construction that has subsequently been replaced in TLS versions [RFC7366]. Unfortunately, the mechanism to correct this flaw relies on extensions: a feature added in TLS 1.0. SSLv3 cannot be updated to correct this flaw in the same way.

The flaws in the CBC modes in SSLv3 are mirrored by the weakness of the stream ciphers it defines. Of those defined, only RC4 is currently in widespread use. RC4, however, exhibits serious biases and is also no longer fit for use [RFC7465].

This leaves SSLv3 with no suitable record protection mechanism.

4.2. Key Exchange

The SSLv3 key exchange is vulnerable to man-in-the-middle attacks when renegotiation [RFC5746] or session resumption [TRIPLE-HS] are used. Each flaw has been fixed in TLS by means of extensions. Again, SSLv3 cannot be updated to correct these flaws.

4.3. Custom Cryptographic Primitives

SSLv3 defines custom constructions for Pseudorandom Function (PRF), Hashed Message Authentication Code (HMAC), and digital signature primitives. Such constructions lack the deep cryptographic scrutiny that standard constructions used by TLS have received. Furthermore, all SSLv3 primitives rely on SHA-1 [RFC3174] and MD5 [RFC1321]: these hash algorithms are considered weak and are being systematically replaced with stronger hash functions, such as SHA-256 [FIPS180-4].

5. Limited Capabilities

SSLv3 is unable to take advantage of the many features that have been added to recent TLS versions. This includes the features that are enabled by ClientHello extensions, which SSLv3 does not support.

Though SSLv3 can benefit from new cipher suites, it cannot benefit from new cryptographic modes and features. Of these, the following are particularly prominent:

- Authenticated Encryption with Additional Data (AEAD) modes are added in [RFC5246].
- Elliptic Curve Diffie-Hellman (ECDH) and Digital Signature Algorithm (ECDSA) are added in [RFC4492].
- Stateless session tickets [RFC5077].
- A datagram mode of operation, DTLS [RFC6347].
- Application-layer protocol negotiation [RFC7301].

6. Security Considerations

This entire document aims to improve security by prohibiting the use of a protocol that is not secure.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2246] Dierks, T. and C. Allen, "The TLS Protocol Version 1.0", RFC 2246, DOI 10.17487/RFC2246, January 1999, <http://www.rfc-editor.org/info/rfc2246>.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, DOI 10.17487/RFC4346, April 2006, <http://www.rfc-editor.org/info/rfc4346>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC6101] Freier, A., Karlton, P., and P. Kocher, "The Secure Sockets Layer (SSL) Protocol Version 3.0", RFC 6101, DOI 10.17487/RFC6101, August 2011, <http://www.rfc-editor.org/info/rfc6101>.

[RFC7366] Gutmann, P., "Encrypt-then-MAC for Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7366, DOI 10.17487/RFC7366, September 2014, <http://www.rfc-editor.org/info/rfc7366>.

[RFC7465] Popov, A., "Prohibiting RC4 Cipher Suites", RFC 7465, DOI 10.17487/RFC7465, February 2015, <http://www.rfc-editor.org/info/rfc7465>.

7.2. Informative References

[FIPS180-4] U.S. National Institute of Standards and Technology, "Secure Hash Standard", FIPS 180-4, March 2012.

[POODLE] Moeller, B., "This POODLE bites: exploiting the SSL 3.0 fallback", October 2014, <http://googleonlinesecurity.blogspot.com/2014/10/this-poodle-bites-exploiting-ssl-30.html>.

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, DOI 10.17487/RFC1321, April 1992, <http://www.rfc-editor.org/info/rfc1321>.

[RFC3174] Eastlake 3rd, D. and P. Jones, "US Secure Hash Algorithm 1 (SHA1)", RFC 3174, DOI 10.17487/RFC3174, September 2001, <http://www.rfc-editor.org/info/rfc3174>.

[RFC4492] Blake-Wilson, S., Bolyard, N., Gupta, V., Hawk, C., and B. Moeller, "Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)", RFC 4492, DOI 10.17487/RFC4492, May 2006, <http://www.rfc-editor.org/info/rfc4492>.

[RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, DOI 10.17487/RFC5077, January 2008, <http://www.rfc-editor.org/info/rfc5077>.

[RFC5746] Rescorla, E., Ray, M., Dispensa, S., and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension", RFC 5746, DOI 10.17487/RFC5746, February 2010, <http://www.rfc-editor.org/info/rfc5746>.

[RFC6176] Turner, S. and T. Polk, "Prohibiting Secure Sockets Layer (SSL) Version 2.0", RFC 6176, DOI 10.17487/RFC6176, March 2011, <http://www.rfc-editor.org/info/rfc6176>.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, DOI 10.17487/RFC6347, January 2012, <http://www.rfc-editor.org/info/rfc6347>.

[RFC7301] Friedl, S., Popov, A., Langley, A., and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", RFC 7301, DOI 10.17487/RFC7301, July 2014, <http://www.rfc-editor.org/info/rfc7301>.

[TRIPLE-HS] Bhargavan, K., Delignat-Lavaud, A., Fournet, C., Pironti, A., and P-Y. Strub, "Triple Handshakes and Cookie Cutters: Breaking and Fixing Authentication over TLS", IEEE Symposium on Security and Privacy, 2014.

Authors' Addresses

Richard Barnes
Mozilla

EMail: rlb@ipv.sx

Martin Thomson
Mozilla

EMail: martin.thomson@gmail.com

Alfredo Pironti
INRIA

EMail: alfredo@pironti.eu

Adam Langley
Google

EMail: agl@google.com