Internet Engineering Task Force (IETF)
Request for Comments: 7457
Category: Informational
ISSN: 2070-1721

Y. Sheffer
Porticor
R. Holz
Technische Universitaet Muenchen
P. Saint-Andre
&yet

February 2015

Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)

Abstract

Over the last few years, there have been several serious attacks on Transport Layer Security (TLS), including attacks on its most commonly used ciphers and modes of operation. This document summarizes these attacks, with the goal of motivating generic and protocol-specific recommendations on the usage of TLS and Datagram TLS (DTLS).

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

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). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see 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/rfc7457.

Table of Contents

1. Introduction

Over the last few years, there have been several major attacks on TLS [RFC5246], including attacks on its most commonly used ciphers and modes of operation. Details are given in Section 2, but a quick summary is that both AES-CBC and RC4, which together make up for most current usage, have been seriously attacked in the context of TLS.

This situation was one of the motivations for the creation of the UTA working group, which was tasked with the creation of generic and protocol-specific recommendations for the use of TLS and DTLS [RFC6347] (unless otherwise noted under Section 3, all of the information provided in this document applies to DTLS).

There is an old saying attributed, ironically enough, to the US National Security Agency (NSA): "Attacks always get better; they never get worse." Unfortunately, that saying is true, so any description of security attacks can only be a snapshot in time. Therefore this document reflects our knowledge as of this writing. It seems likely that new attacks will be discovered in the future.

For a more detailed discussion of the attacks listed here, the interested reader is referred to [Attacks-iSec].

2. Attacks on TLS

This section lists the attacks that motivated the current recommendations in [SECURE-TLS]. This list is not intended to be an extensive survey of the security of TLS.

While there are widely deployed mitigations for some of the attacks listed below, we believe that their root causes necessitate a more systematic solution, which we have attempted to develop in [SECURE-TLS].

When an identifier exists for an attack, we have included its Common Vulnerabilities and Exposures (CVE) ID. CVE [CVE] is an extensive, industry-wide database of software vulnerabilities.

2.1. SSL Stripping

Various attacks attempt to remove the use of Secure Socket Layer / Transport Layer Security (SSL/TLS) altogether by modifying unencrypted protocols that request the use of TLS, specifically modifying HTTP traffic and HTML pages as they pass on the wire. These attacks are known collectively as "SSL Stripping" (a form of the more generic "downgrade attack") and were first introduced by Moxie Marlinspike [SSL-Stripping]. In the context of Web traffic,

these attacks are only effective if the client initially accesses a Web server using HTTP. A commonly used mitigation is HTTP Strict Transport Security (HSTS) [RFC6797].

2.2. STARTTLS Command Injection Attack (CVE-2011-0411)

Similarly, there are attacks on the transition between unprotected and TLS-protected traffic. A number of IETF application protocols have used an application-level command, usually STARTTLS, to upgrade a cleartext connection to use TLS. Multiple implementations of STARTTLS had a flaw where an application-layer input buffer retained commands that were pipelined with the STARTTLS command, such that commands received prior to TLS negotiation are executed after TLS negotiation. This problem is resolved by requiring the application-level command input buffer to be empty before negotiating TLS. Note that this flaw lives in the application layer code and does not impact the TLS protocol directly.

STARTTLS and similar mechanisms are vulnerable to downgrade attacks, whereby the attacker simply removes the STARTTLS indication from the (unprotected) request. This cannot be mitigated unless HSTS-like solutions are added.

2.3. BEAST (CVE-2011-3389)

The BEAST attack [BEAST] uses issues with the TLS 1.0 implementation of Cipher Block Chaining (CBC) (that is, the predictable initialization vector) to decrypt parts of a packet, and specifically to decrypt HTTP cookies when HTTP is run over TLS.

2.4. Padding Oracle Attacks

A consequence of the MAC-then-encrypt design in all current versions of TLS is the existence of padding oracle attacks [Padding-Oracle]. A recent incarnation of these attacks is the Lucky Thirteen attack (CVE-2013-0169) [CBC-Attack], a timing side-channel attack that allows the attacker to decrypt arbitrary ciphertext.

The Lucky Thirteen attack can be mitigated by using authenticated encryption like AES-GCM [RFC5288] or encrypt-then-MAC [RFC7366] instead of the TLS default of MAC-then-encrypt.

An even newer variant of the padding oracle attack, one that does not use timing information, is the POODLE attack (CVE-2014-3566) [POODLE] on SSL 3.0. This attack has no known mitigation.

2.5. Attacks on RC4

The RC4 algorithm [RC4] has been used with TLS (and previously, SSL) for many years. RC4 has long been known to have a variety of cryptographic weaknesses, e.g., [RC4-Attack-Pau], [RC4-Attack-Man], and [RC4-Attack-FMS]. Recent cryptanalysis results [RC4-Attack-AlF] exploit biases in the RC4 keystream to recover repeatedly encrypted plaintexts.

These recent results are on the verge of becoming practically exploitable; currently they require 2^26 sessions or 13x2^30 encryptions. As a result, RC4 can no longer be seen as providing a sufficient level of security for TLS sessions. For further details, the reader is referred to [CIPHER-SUITES] and the references it cites.

2.6. Compression Attacks: CRIME, TIME, and BREACH

The CRIME attack [CRIME] (CVE-2012-4929) allows an active attacker to decrypt ciphertext (specifically, cookies) when TLS is used with TLS-level compression.

The TIME attack [TIME] and the later BREACH attack [BREACH] (CVE-2013-3587, though the number has not been officially allocated) both make similar use of HTTP-level compression to decrypt secret data passed in the HTTP response. We note that compression of the HTTP message body is much more prevalent than compression at the TLS level.

The TIME attack can be mitigated by disabling TLS compression. We are not aware of mitigations at the TLS protocol level to the BREACH attack, and so application-level mitigations are needed (see [BREACH]). For example, implementations of HTTP that use Cross-Site Request Forgery (CSRF) tokens will need to randomize them. Even the best practices and recommendations from [SECURE-TLS] are insufficient to thwart this attack.

2.7. Certificate and RSA-Related Attacks

There have been several practical attacks on TLS when used with RSA certificates (the most common use case). These include [Bleichenbacher98] and [Klima03]. While the Bleichenbacher attack has been mitigated in TLS 1.0, the Klima attack, which relies on a version-check oracle, is only mitigated by TLS 1.1.

The use of RSA certificates often involves exploitable timing issues [Brumley03] (CVE-2003-0147), unless the implementation takes care to explicitly eliminate them.

A recent certificate fuzzing tool [Brubaker2014using] uncovered numerous vulnerabilities in different TLS libraries related to certificate validation.

2.8. Theft of RSA Private Keys

When TLS is used with most non-Diffie-Hellman cipher suites, it is sufficient to obtain the server's private key in order to decrypt any sessions (past and future) that were initiated with that server. This technique is used, for example, by the popular Wireshark network sniffer to inspect TLS-protected connections.

It is known that stolen (or otherwise obtained) private keys have been used as part of large-scale monitoring [RFC7258] of certain servers.

Such attacks can be mitigated by better protecting the private key, e.g., using OS protections or dedicated hardware. Even more effective is the use of cipher suites that offer "forward secrecy", the property where revealing a secret such as a private key does not expose past or future sessions to a passive attacker.

2.9. Diffie-Hellman Parameters

TLS allows the definition of ephemeral Diffie-Hellman (DH) and Elliptic Curve Diffie-Hellman parameters in its respective key exchange modes. This results in an attack detailed in [Cross-Protocol]. Using predefined DH groups, as proposed in [FFDHE-TLS], would mitigate this attack.

In addition, clients that do not properly verify the received parameters are exposed to man-in-the-middle (MITM) attacks. Unfortunately, the TLS protocol does not mandate this verification (see [RFC6989] for analogous information for IPsec).

2.10. Renegotiation (CVE-2009-3555)

A major attack on the TLS renegotiation mechanism applies to all current versions of the protocol. The attack and the TLS extension that resolves it are described in [RFC5746].

2.11. Triple Handshake (CVE-2014-1295)

The triple handshake attack [BhargavanDFPS14] enables the attacker to cause two TLS connections to share keying material. This leads to a multitude of attacks, e.g., man-in-the-middle, breaking safe renegotiation, and breaking channel binding via TLS Exporter [RFC5705] or "tls-unique" [RFC5929].

2.12. Virtual Host Confusion

A recent article [Delignat14] describes a security issue whereby SSLv3 fallback and improper handling of session caches on the server side can be abused by an attacker to establish a malicious connection to a virtual host other than the one originally intended and approved by the server. This attack is especially serious in performance critical environments where sharing of SSLv3 session caches is very common.

2.13. Denial of Service

Server CPU power has progressed over the years so that TLS can now be turned on by default. However, the risk of malicious clients and coordinated groups of clients ("botnets") mounting denial-of-service attacks is still very real. TLS adds another vector for computational attacks, since a client can easily (with little computational effort) force the server to expend relatively large computational work. It is known that such attacks have in fact been mounted.

2.14. Implementation Issues

Even when the protocol is properly specified, this does not guarantee the security of implementations. In fact, there are very common issues that often plague TLS implementations. In particular, when integrating into higher-level protocols, TLS and its PKI-based authentication are sometimes the source of misunderstandings and implementation "shortcuts". An extensive survey of these issues can be found in [Georgiev2012].

- Implementations might omit validation of the server certificate altogether. For example, this is true of the default implementation of HTTP client libraries in Python 2 (e.g., CVE-2013-2191).

- Implementations might not validate the server identity. This validation typically amounts to matching the protocol-level server name with the certificate's Subject Alternative Name field. Note: this same information is often also found in the Common Name part of the Distinguished Name, and some validators incorrectly retrieve it from there instead of from the Subject Alternative Name.

- Implementations might validate the certificate chain incorrectly or not at all, or use an incorrect or outdated trust anchor list.

An implementation attack of a different kind, one that exploits a simple coding mistake (bounds check), is the Heartbleed attack (CVE-2014-0160) that affected a wide swath of the Internet when it was discovered in April 2014.

2.15. Usability

Many TLS endpoints, such as browsers and mail clients, allow the user to explicitly accept an invalid server certificate. This often takes the form of a UI dialog (e.g., "do you accept this server?"), and users have been conditioned to respond in the affirmative in order to allow the connection to take place.

This user behavior is used by (arguably legitimate) "SSL proxies" that decrypt and re-encrypt the TLS connection in order to enforce local security policy. It is also abused by attackers whose goal is to gain access to the encrypted information.

Mitigation is complex and will probably involve a combination of protocol mechanisms (HSTS, certificate pinning [KEY-PINNING]), and very careful UI design.

3. Applicability to DTLS

DTLS [RFC4347] [RFC6347] is an adaptation of TLS for UDP.

With respect to the attacks described in the current document, DTLS 1.0 is equivalent to TLS 1.1. The only exception is RC4, which is disallowed in DTLS. DTLS 1.2 is equivalent to TLS 1.2.

4. Security Considerations

This document describes protocol attacks in an informational manner and in itself does not have any security implications. Its companion documents, especially [SECURE-TLS], certainly do.

5. Informative References

[Attacks-iSec] Sarkar, P. and S. Fitzgerald, "Attacks on SSL, a comprehensive study of BEAST, CRIME, TIME, BREACH, Lucky13 and RC4 biases", August 2013, <https://www.isecpartners.com/media/106031/ssl_attacks_survey.pdf>.

[BEAST] Rizzo, J. and T. Duong, "Browser Exploit Against SSL/TLS", 2011, <http://packetstormsecurity.com/files/105499/Browser-Exploit-Against-SSL-TLS.html>.

[BREACH] Prado, A., Harris, N., and Y. Gluck, "The BREACH Attack", 2013, <http://breachattack.com/>.

[BhargavanDFPS14]
Bhargavan, K., Delignat-Lavaud, A., Fournet, C., Pironti, A., and P. Strub, "Triple handshakes and cookie cutters: breaking and fixing authentication over tls", 2014, <https://secure-resumption.com/tlsauth.pdf>.

[Bleichenbacher98]
Bleichenbacher, D., "Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1", 1998, <http://archiv.infsec.ethz.ch/education/fs08/secsem/Bleichenbacher98.pdf>.

[Brubaker2014using]
Brubaker, C., Jana, S., Ray, B., Khurshid, S., and V. Shmatikov, "Using Frankencerts for Automated Adversarial Testing of Certificate Validation in SSL/TLS Implementations", 2014, <https://www.cs.utexas.edu/~shmat/shmat_oak14.pdf>.

[Brumley03]
Brumley, D. and D. Boneh, "Remote Timing Attacks are Practical", 2003, <http://crypto.stanford.edu/~dabo/papers/ssl-timing.pdf>.

[CBC-Attack]
AlFardan, N. and K. Paterson, "Lucky Thirteen: Breaking the TLS and DTLS Record Protocols", IEEE Symposium on Security and Privacy, 2013, <http://www.ieee-security.org/TC/SP2013/papers/4977a526.pdf>.

[CIPHER-SUITES]
Popov, A., "Prohibiting RC4 Cipher Suites", Work in Progress, draft-ietf-tls-prohibiting-rc4-01, October 2014.

[CRIME] Rizzo, J. and T. Duong, "The CRIME Attack", EKOparty Security Conference, 2012.

[CVE] MITRE, "Common Vulnerabilities and Exposures", <https://cve.mitre.org/>.

[Cross-Protocol]
Mavrogiannopoulos, N., Vercauteren, F., Velichkov, V., and B. Preneel, "A cross-protocol attack on the TLS protocol", Proceedings of the 2012 ACM Conference in Computer and Communications Security, pages 62-72, 2012, <http://doi.acm.org/10.1145/2382196.2382206>.

[Delignat14]
Delignat-Lavaud, A. and K. Bhargavan, "Virtual Host Confusion: Weaknesses and Exploits", Black Hat 2014, 2014, <https://bh.ht.vc/vhost_confusion.pdf>.

[FFDHE-TLS]
Gillmor, D., "Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for TLS", Work in Progress, draft-ietf-tls-negotiated-ff-dhe-05, December 2014.

[Georgiev2012]
Georgiev, M., Iyengar, S., Jana, S., Anubhai, R., Boneh, D., and V. Shmatikov, "The most dangerous code in the world: validating SSL certificates in non-browser software", Proceedings of the 2012 ACM conference on Computer and Communications Security, pages 38-49, 2012, <http://doi.acm.org/10.1145/2382196.2382204>.

[KEY-PINNING]
Evans, C., Palmer, C., and R. Sleevi, "Public Key Pinning Extension for HTTP", Work in Progress, draft-ietf-websec-key-pinning-21, October 2014.

[Klima03] Klima, V., Pokorny, O., and T. Rosa, "Attacking RSA-based Sessions in SSL/TLS", 2003, <https://eprint.iacr.org/2003/052.pdf>.

[POODLE] Moeller, B., Duong, T., and K. Kotowicz, "This POODLE Bites: Exploiting the SSL 3.0 Fallback", September 2014, <https://www.openssl.org/~bodo/ssl-poodle.pdf>.

[Padding-Oracle]
Vaudenay, S., "Security Flaws Induced by CBC Padding Applications to SSL, IPSEC, WTLS...", EUROCRYPT 2002, 2002, <http://www.iacr.org/cryptodb/archive/2002/EUROCRYPT/2850/2850.pdf>.

[RC4] Schneier, B., "Applied Cryptography: Protocols, Algorithms, and Source Code in C", Second Edition, October 1996.

[RC4-Attack-AlF]
AlFardan, N., Bernstein, D., Paterson, K., Poettering, B., and J. Schuldt, "On the Security of RC4 in TLS", Usenix Security Symposium 2013, August 2013, <https://www.usenix.org/conference/usenixsecurity13/security-rc4-tls>.

[RC4-Attack-FMS]
Fluhrer, S., Mantin, I., and A. Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4", Selected Areas in Cryptography, August 2001, <http://www.crypto.com/papers/others/rc4_ksaproc.pdf>.

[RC4-Attack-Man]
Mantin, I. and A. Shamir, "A Practical Attack on Broadcast RC4", April 2001, <http://saluc.engr.uconn.edu/refs/stream_cipher/mantin01attackRC4.pdf>.

[RC4-Attack-Pau]
Paul, G. and S. Maitra, "Permutation After RC4 Key Scheduling Reveals the Secret Key", August 2007, <http://dblp.uni-trier.de/db/conf/sacrypt/sacrypt2007.html#PaulM07>.

[RFC4347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security", RFC 4347, April 2006, <http://www.rfc-editor.org/info/rfc4347>.

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

[RFC5288] Salowey, J., Choudhury, A., and D. McGrew, "AES Galois Counter Mode (GCM) Cipher Suites for TLS", RFC 5288, August 2008, <http://www.rfc-editor.org/info/rfc5288>.

[RFC5705] Rescorla, E., "Keying Material Exporters for Transport Layer Security (TLS)", RFC 5705, March 2010, <http://www.rfc-editor.org/info/rfc5705>.

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

[RFC5929] Altman, J., Williams, N., and L. Zhu, "Channel Bindings for TLS", RFC 5929, July 2010, <http://www.rfc-editor.org/info/rfc5929>.

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

[RFC6797] Hodges, J., Jackson, C., and A. Barth, "HTTP Strict Transport Security (HSTS)", RFC 6797, November 2012, <http://www.rfc-editor.org/info/rfc6797>.

[RFC6989] Sheffer, Y. and S. Fluhrer, "Additional Diffie-Hellman Tests for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 6989, July 2013, <http://www.rfc-editor.org/info/rfc6989>.

[RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, May 2014, <http://www.rfc-editor.org/info/rfc7258>.

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

[SECURE-TLS] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of TLS and DTLS", Work in Progress, draft-ietf-uta-tls-bcp-08, December 2014.

[SSL-Stripping] Marlinspike, M., "sslstrip", February 2009, <http://www.thoughtcrime.org/software/sslstrip/>.

[TIME] Be'ery, T. and A. Shulman, "A Perfect CRIME? Only TIME Will Tell", Black Hat Europe 2013, 2013, <https://media.blackhat.com/eu-13/briefings/Beery/bh-eu-13-a-perfect-crime-beery-wp.pdf>.

Acknowledgements

We would like to thank Stephen Farrell, Simon Josefsson, John Mattsson, Yoav Nir, Kenny Paterson, Patrick Pelletier, Tom Ritter, Rich Salz, and Meral Shirazipour for their feedback on this document. We thank Andrei Popov for contributing text on RC4, Kohei Kasamatsu for text on Lucky13, Ilari Liusvaara for text on attacks and on DTLS, Aaron Zauner for text on virtual host confusion, and Chris Newman for text on STARTTLS command injection. Ralph Holz gratefully acknowledges the support of NICTA (National ICT of Australia) in the preparation of this document.

During IESG review, Richard Barnes, Barry Leiba, and Kathleen Moriarty caught several issues that needed to be addressed.

The authors gratefully acknowledge the assistance of Leif Johansson and Orit Levin as the working group chairs and Pete Resnick as the sponsoring Area Director.

The document was prepared using the lyx2rfc tool, created by Nico Williams.

Authors' Addresses

Yaron Sheffer
Porticor
29 HaHarash St.
Hod HaSharon 4501303
Israel

EMail: yaronf.ietf@gmail.com

Ralph Holz
Technische Universitaet Muenchen
Boltzmannstr. 3
Garching 85748
Germany

EMail: holz@net.in.tum.de

Peter Saint-Andre
&yet

EMail: peter@andyet.com
URI: https://andyet.com/