TLS Working Group
Internet-Draft
Updates: 9325, 4346, 5246, 4162, 6347, 5932, 5288, 6209, 6367, 8422, 5289, 5469, 4785, 4279, 5487, 6655, 7905 (if approved)
Intended status: Standards Track
Expires: 16 May 2026

N. Aviram
12 November 2025

Deprecating Obsolete Key Exchange Methods in (D)TLS 1.2
draft-ietf-tls-deprecate-obsolete-kex-07

Abstract

For (D)TLS 1.2, this document deprecates the use of two key exchanges, namely Diffie-Hellman over a finite field and RSA, and it discourages the use of static elliptic curve Diffie-Hellman cipher suites.

These prescriptions apply only to (D)TLS 1.2 since (D)TLS 1.0 and TLS 1.1 are deprecated by RFC 8996 and (D)TLS 1.3 either does not use the affected algorithm or does not share the relevant configuration options. (There is no DTLS version 1.1.)

This document updates RFCs 9325, 4346, 5246, 4162, 6347, 5932, 5288, 6209, 6367, 8422, 5289, 5469, 4785, 4279, 5487, 6655, and 7905, to deprecate or discourage - i.e., change to MUST NOT or SHOULD NOT, as listed in Section 5.3 Section 5.2 Section 5.3 Section 5.4 Section 5.5 - the use of cipher suites using the above key exchange methods in (D)TLS 1.2 connections.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

(D)TLS 1.2 supports a variety of key exchange algorithms, including RSA, Diffie-Hellman over a finite field, and elliptic curve Diffie-Hellman (ECDH).

Diffie-Hellman key exchange, over any group, comes in ephemeral and non-ephemeral varieties. Non-ephemeral DH algorithms use static DH public keys included in the authenticating peer's certificate; see

[RFC4492] for discussion. In contrast, ephemeral DH algorithms use ephemeral DH public keys sent in the handshake and authenticated by the peer’s certificate. Ephemeral and non-ephemeral finite field DH algorithms are called DHE and DH (or FFDHE and FFDH), respectively, and ephemeral and non-ephemeral elliptic curve DH algorithms are called ECDHE and ECDH, respectively [RFC4492].

In general, non-ephemeral cipher suites are not recommended due to their lack of forward secrecy. Moreover, as demonstrated by the [Raccoon] attack on finite-field DH, public key reuse, either via non-ephemeral cipher suites or reused keys with ephemeral cipher suites, can lead to timing side channels that may leak connection secrets. For elliptic curve DH, invalid curve attacks similarly exploit secret reuse in order to break security [ICA], further demonstrating the risk of reusing public keys. While both side channels can be avoided in implementations, experience shows that in practice, implementations may fail to thwart such attacks due to the complexity and number of the required mitigations.

Additionally, RSA key exchange suffers from security problems that are independent of implementation choices as well as problems that stem purely from the difficulty of implementing security countermeasures correctly.

At a rough glance, the problems affecting FFDHE in (D)TLS 1.2 are as follows:

1. FFDHE suffers from interoperability problems because there is no mechanism for negotiating the group, and some implementations only support small group sizes (see [RFC7919], Section 1).

2. FFDHE groups may have small subgroups, which enables several attacks [subgroups]. When presented with a custom, non-standardized FFDHE group, a handshaking client cannot practically verify that the group chosen by the server does not suffer from this problem. There is also no mechanism for such handshakes to fall back to other key exchange parameters that are acceptable to the client. Custom FFDHE groups are widespread (as a result of advice based on [weak-dh]). Therefore, clients cannot simply reject handshakes that present custom, and thus potentially dangerous, groups.

3. In practice, some operators use 1024-bit FFDHE groups since this is the maximum size that ensures wide support (see [RFC7919], Section 1). This size leaves only a small security margin vs. the current discrete log record, which stands at 795 bits [DLOG795].

4. Expanding on the previous point, just a handful of very large computations allow an attacker to cheaply decrypt a relatively large fraction of FFDHE traffic (namely, traffic encrypted using particular standardized groups) [weak-dh].

5. When secrets are not fully ephemeral, FFDHE suffers from the [Raccoon] side channel attack. (Note that FFDH is inherently vulnerable to the Raccoon attack unless constant-time mitigations are employed.)

The problems affecting RSA key exchange in (D)TLS 1.2 are as follows:

1. RSA key exchange offers no forward secrecy, by construction.

2. RSA key exchange may be vulnerable to Bleichenbacher's attack [BLEI]. Experience shows that variants of this attack arise every few years because implementing the relevant countermeasure correctly is difficult (see [ROBOT], [NEW-BLEI], [DROWN]).

3. In addition to the above point, there is no convenient mechanism in (D)TLS 1.2 for the domain separation of keys. Therefore, a single endpoint that is vulnerable to Bleichenbacher's attack would affect all endpoints sharing the same RSA key (see [XPROT], [DROWN]).

This document updates [RFC9325], [RFC4346], [RFC5246], [RFC4162], [RFC6347], [RFC5932], [RFC5288], [RFC6209], [RFC6367], [RFC8422], [RFC5289], [RFC4785], [RFC4279], [RFC5487], [RFC6655], [RFC7905] and [RFC5469] to remediate the above problems, by deprecating and discouraging the use of affected cipher suites, as listed in Section 5.3 Section 5.2 Section 5.3 Section 5.4 Section 5.5.

BCP 195 [RFC8996][RFC9325] contains the latest IETF recommendations for users of the (D)TLS protocol (and specifically, (D)TLS 1.2) and this document supersedes it in several points. Section 6 details the exact differences. All other recommendations of the BCP document remain valid.

1.1. Requirements

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Non-Ephemeral Diffie-Hellman

Clients MUST NOT offer and servers MUST NOT select non-ephemeral FFDH cipher suites in (D)TLS 1.2 connections. (Note that (D)TLS 1.0 and TLS 1.1 are deprecated by [RFC8996] and (D)TLS 1.3 does not support FFDH [I-D.ietf-tls-rfc8446bis][RFC9147].) This includes all cipher suites listed in the table in Section 5.1.

Clients SHOULD NOT offer and servers SHOULD NOT select non-ephemeral ECDH cipher suites in (D)TLS 1.2 connections. (This requirement is already present in [RFC9325]. Note that (D)TLS 1.0 and TLS 1.1 are deprecated by [RFC8996] and (D)TLS 1.3 does not support ECDH [I-D.ietf-tls-rfc8446bis][RFC9147].) This includes all cipher suites listed in the table in Section 5.2.

In addition, to avoid the use of non-ephemeral Diffie-Hellman, clients SHOULD NOT use and servers SHOULD NOT accept certificates with fixed DH parameters. These certificate types are rsa_fixed_dh, dss_fixed_dh, rsa_fixed_ecdh and ecdsa_fixed_ecdh as listed in Section 5.5. These values only apply to (D)TLS versions of 1.2 and below.

3. Ephemeral Finite Field Diffie-Hellman

Clients MUST NOT offer and servers MUST NOT select FFDHE cipher suites in (D)TLS 1.2 connections. This includes all cipher suites listed in the table in Section 5.3. (Note that (D)TLS 1.0 and TLS 1.1 are deprecated by [RFC8996].) FFDHE cipher suites in (D)TLS 1.3 do not suffer from the problems presented in Section 1; see [I-D.ietf-tls-rfc8446bis] and [RFC9147]. Therefore, clients and servers MAY offer FFDHE cipher suites in (D)TLS 1.3 connections.

4. RSA

Clients MUST NOT offer and servers MUST NOT select RSA cipher suites in (D)TLS 1.2 connections. (Note that (D)TLS 1.0 and TLS 1.1 are deprecated by [RFC8996], and (D)TLS 1.3 does not support static RSA [I-D.ietf-tls-rfc8446bis][RFC9147].) This includes all cipher suites listed in the table in Section 5.4. Note that these cipher suites are already marked as not recommended in the "TLS Cipher Suites" registry [tls-registry].

5. Updates to Cipher Suites and TLS ClientCertificateType Identifiers

5.1. DH Cipher Suites Deprecated by This Document

This document requests IANA to set the “recommended” column to “D” for the following entries:

Ciphersuite	Reference
TLS_DH_DSS_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_DH_DSS_WITH_DES_CBC_SHA	[RFC5469]
TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_DH_RSA_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_DH_RSA_WITH_DES_CBC_SHA	[RFC5469]
TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_DH_anon_EXPORT_WITH_RC4_40_MD5	[RFC4346][RFC6347]
TLS_DH_anon_WITH_RC4_128_MD5	[RFC5246][RFC6347]
TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_DH_anon_WITH_DES_CBC_SHA	[RFC5469]
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_DH_DSS_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_DH_RSA_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_DH_anon_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_DH_DSS_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_DH_RSA_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_DH_anon_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_DH_DSS_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_DH_RSA_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_DH_DSS_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_DH_anon_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_DH_DSS_WITH_AES_256_CBC_SHA256	[RFC5246]

TLS_DH_RSA_WITH_AES_256_CBC_SHA256	[RFC5246]
TLS_DH_anon_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_DH_anon_WITH_AES_256_CBC_SHA256	[RFC5246]
TLS_DH_DSS_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_DH_anon_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_DH_DSS_WITH_SEED_CBC_SHA	[RFC4162]
TLS_DH_RSA_WITH_SEED_CBC_SHA	[RFC4162]
TLS_DH_anon_WITH_SEED_CBC_SHA	[RFC4162]
TLS_DH_RSA_WITH_AES_128_GCM_SHA256	[RFC5288]
TLS_DH_RSA_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_DH_DSS_WITH_AES_128_GCM_SHA256	[RFC5288]
TLS_DH_DSS_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_DH_anon_WITH_AES_128_GCM_SHA256	[RFC5288]
TLS_DH_anon_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_DH_DSS_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_DH_RSA_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_DH_anon_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_DH_DSS_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_DH_RSA_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_DH_anon_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_DH_DSS_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_DH_DSS_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DH_RSA_WITH_ARIA_128_CBC_SHA256	[RFC6209]

TLS_DH_RSA_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DH_anon_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_DH_anon_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DH_RSA_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_DH_RSA_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DH_DSS_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_DH_DSS_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DH_anon_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_DH_anon_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DH_RSA_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DH_RSA_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_DH_DSS_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DH_DSS_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_DH_anon_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DH_anon_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]

Table 1

5.2. ECDH Cipher Suites Whose Use Is Discouraged by This Document

[RFC9325] already specifies that implementations SHOULD NOT negotiate the following cipher suites; accordingly, they appear with "Recommended: N" in the IANA TLS Cipher Suites registry [tls-registry]. This document updates them to "Recommended: D" to align with [I-D.ietf-tls-rfc8447bis]. It also records the rationale for discouraging use of these cipher suites, and cites prior analyses and attacks that demonstrate the associated risks.

Ciphersuite	Reference
TLS_ECDH_ECDSA_WITH_NULL_SHA	[RFC8422]
TLS_ECDH_ECDSA_WITH_RC4_128_SHA	[RFC8422][RFC6347]
TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA	[RFC8422]
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	[RFC8422]
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	[RFC8422]
TLS_ECDH_RSA_WITH_NULL_SHA	[RFC8422]
TLS_ECDH_RSA_WITH_RC4_128_SHA	[RFC8422][RFC6347]
TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA	[RFC8422]
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA	[RFC8422]
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	[RFC8422]
TLS_ECDH_anon_WITH_NULL_SHA	[RFC8422]
TLS_ECDH_anon_WITH_RC4_128_SHA	[RFC8422][RFC6347]
TLS_ECDH_anon_WITH_3DES_EDE_CBC_SHA	[RFC8422]
TLS_ECDH_anon_WITH_AES_128_CBC_SHA	[RFC8422]
TLS_ECDH_anon_WITH_AES_256_CBC_SHA	[RFC8422]
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	[RFC5289]
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	[RFC5289]
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	[RFC5289]
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384	[RFC5289]
TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	[RFC5289]
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384	[RFC5289]
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256	[RFC5289]
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384	[RFC5289]

TLS_ECDH_ECDSA_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_ECDH_ECDSA_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_ECDH_RSA_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_ECDH_RSA_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_ECDH_ECDSA_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_ECDH_ECDSA_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_ECDH_RSA_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_ECDH_RSA_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_ECDH_ECDSA_WITH_CAMELLIA_128_CBC_SHA256	[RFC6367]
TLS_ECDH_ECDSA_WITH_CAMELLIA_256_CBC_SHA384	[RFC6367]
TLS_ECDH_RSA_WITH_CAMELLIA_128_CBC_SHA256	[RFC6367]
TLS_ECDH_RSA_WITH_CAMELLIA_256_CBC_SHA384	[RFC6367]
TLS_ECDH_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_ECDH_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_ECDH_RSA_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_ECDH_RSA_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]

Table 2

5.3. DHE Cipher Suites deprecated by This Document

This document requests IANA to set the “recommended” column to “D” for the following entries:

Ciphersuite	Reference
TLS_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_DHE_DSS_WITH_DES_CBC_SHA	[RFC5469] [RFC8996]

TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_DHE_RSA_WITH_DES_CBC_SHA	[RFC5469] [RFC8996]
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_DHE_PSK_WITH_NULL_SHA	[RFC4785]
TLS_DHE_DSS_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_DHE_DSS_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256	[RFC5246]
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	[RFC5246]
TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_DHE_PSK_WITH_RC4_128_SHA	[RFC4279][RFC6347]
TLS_DHE_PSK_WITH_3DES_EDE_CBC_SHA	[RFC4279]
TLS_DHE_PSK_WITH_AES_128_CBC_SHA	[RFC4279]
TLS_DHE_PSK_WITH_AES_256_CBC_SHA	[RFC4279]
TLS_DHE_DSS_WITH_SEED_CBC_SHA	[RFC4162]
TLS_DHE_RSA_WITH_SEED_CBC_SHA	[RFC4162]
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	[RFC5288]

TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	[RFC5288]
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_DHE_PSK_WITH_AES_128_GCM_SHA256	[RFC5487]
TLS_DHE_PSK_WITH_AES_256_GCM_SHA384	[RFC5487]
TLS_DHE_PSK_WITH_AES_128_CBC_SHA256	[RFC5487]
TLS_DHE_PSK_WITH_AES_256_CBC_SHA384	[RFC5487]
TLS_DHE_PSK_WITH_NULL_SHA256	[RFC5487]
TLS_DHE_PSK_WITH_NULL_SHA384	[RFC5487]
TLS_DHE_DSS_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_DHE_DSS_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_DHE_DSS_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_DHE_DSS_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DHE_RSA_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_DHE_RSA_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DHE_RSA_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_DHE_RSA_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DHE_DSS_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_DHE_DSS_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DHE_PSK_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_DHE_PSK_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_DHE_PSK_WITH_ARIA_128_GCM_SHA256	[RFC6209]

TLS_DHE_PSK_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_DHE_DSS_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DHE_DSS_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_DHE_PSK_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_DHE_PSK_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_DHE_PSK_WITH_CAMELLIA_128_CBC_SHA256	[RFC6367]
TLS_DHE_PSK_WITH_CAMELLIA_256_CBC_SHA384	[RFC6367]
TLS_DHE_RSA_WITH_AES_128_CCM	[RFC6655]
TLS_DHE_RSA_WITH_AES_256_CCM	[RFC6655]
TLS_DHE_RSA_WITH_AES_128_CCM_8	[RFC6655]
TLS_DHE_RSA_WITH_AES_256_CCM_8	[RFC6655]
TLS_DHE_PSK_WITH_AES_128_CCM	[RFC6655]
TLS_DHE_PSK_WITH_AES_256_CCM	[RFC6655]
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	[RFC7905]
TLS_DHE_PSK_WITH_CHACHA20_POLY1305_SHA256	[RFC7905]
TLS_PSK_DHE_WITH_AES_128_CCM_8	[RFC6655]
TLS_PSK_DHE_WITH_AES_256_CCM_8	[RFC6655]

Table 3

5.4. RSA Cipher Suites Deprecated by This Document

This document requests IANA to set the “recommended” column to “D” for the following entries:

Ciphersuite	Reference
TLS_RSA_WITH_NULL_MD5	[RFC5246]
TLS_RSA_WITH_NULL_SHA	[RFC5246]
TLS_RSA_EXPORT_WITH_RC4_40_MD5	[RFC4346][RFC6347]
TLS_RSA_WITH_RC4_128_MD5	[RFC5246][RFC6347]
TLS_RSA_WITH_RC4_128_SHA	[RFC5246][RFC6347]
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5	[RFC4346]
TLS_RSA_WITH_IDEA_CBC_SHA	[RFC5469] [RFC8996]
TLS_RSA_EXPORT_WITH_DES40_CBC_SHA	[RFC4346]
TLS_RSA_WITH_DES_CBC_SHA	[RFC5469] [RFC8996]
TLS_RSA_WITH_3DES_EDE_CBC_SHA	[RFC5246]
TLS_RSA_PSK_WITH_NULL_SHA	[RFC4785]
TLS_RSA_WITH_AES_128_CBC_SHA	[RFC5246]
TLS_RSA_WITH_AES_256_CBC_SHA	[RFC5246]
TLS_RSA_WITH_NULL_SHA256	[RFC5246]
TLS_RSA_WITH_AES_128_CBC_SHA256	[RFC5246]
TLS_RSA_WITH_AES_256_CBC_SHA256	[RFC5246]
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA	[RFC5932]
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA	[RFC5932]
TLS_RSA_PSK_WITH_RC4_128_SHA	[RFC4279][RFC6347]
TLS_RSA_PSK_WITH_3DES_EDE_CBC_SHA	[RFC4279]
TLS_RSA_PSK_WITH_AES_128_CBC_SHA	[RFC4279]
TLS_RSA_PSK_WITH_AES_256_CBC_SHA	[RFC4279]
TLS_RSA_WITH_SEED_CBC_SHA	[RFC4162]

TLS_RSA_WITH_AES_128_GCM_SHA256	[RFC5288]
TLS_RSA_WITH_AES_256_GCM_SHA384	[RFC5288]
TLS_RSA_PSK_WITH_AES_128_GCM_SHA256	[RFC5487]
TLS_RSA_PSK_WITH_AES_256_GCM_SHA384	[RFC5487]
TLS_RSA_PSK_WITH_AES_128_CBC_SHA256	[RFC5487]
TLS_RSA_PSK_WITH_AES_256_CBC_SHA384	[RFC5487]
TLS_RSA_PSK_WITH_NULL_SHA256	[RFC5487]
TLS_RSA_PSK_WITH_NULL_SHA384	[RFC5487]
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256	[RFC5932]
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256	[RFC5932]
TLS_RSA_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_RSA_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_RSA_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_RSA_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_RSA_PSK_WITH_ARIA_128_CBC_SHA256	[RFC6209]
TLS_RSA_PSK_WITH_ARIA_256_CBC_SHA384	[RFC6209]
TLS_RSA_PSK_WITH_ARIA_128_GCM_SHA256	[RFC6209]
TLS_RSA_PSK_WITH_ARIA_256_GCM_SHA384	[RFC6209]
TLS_RSA_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_RSA_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_RSA_PSK_WITH_CAMELLIA_128_GCM_SHA256	[RFC6367]
TLS_RSA_PSK_WITH_CAMELLIA_256_GCM_SHA384	[RFC6367]
TLS_RSA_PSK_WITH_CAMELLIA_128_CBC_SHA256	[RFC6367]
TLS_RSA_PSK_WITH_CAMELLIA_256_CBC_SHA384	[RFC6367]

TLS_RSA_WITH_AES_128_CCM	[RFC6655]
TLS_RSA_WITH_AES_256_CCM	[RFC6655]
TLS_RSA_WITH_AES_128_CCM_8	[RFC6655]
TLS_RSA_WITH_AES_256_CCM_8	[RFC6655]
TLS_RSA_PSK_WITH_CHACHA20_POLY1305_SHA256	[RFC7905]

Table 4

5.5. TLS ClientCertificateType Identifiers Deprecated by This Document

This document requests IANA to set the “recommended” column to “D” for the following entries:

Certificate Type	Reference
rsa_fixed_dh (3)	[RFC5246]
dss_fixed_dh (4)	[RFC5246]
rsa_fixed_ecdh (65)	[RFC8422]
ecdsa_fixed_ecdh (66)	[RFC8422]

Table 5

6. Updating RFC 9325

This document updates [RFC9325] with respect to the use of (D)TLS 1.2, and the table below lists the exact changes. All changes should be made in Section 4.1 of [RFC9325].

// Note to RFC Editor: please replace XXX below by the current RFC
// number.

	RFC 9325	RFC XXX
Non-ephemeral FFDH	SHOULD NOT	MUST NOT
Non-ephemeral ECDH	SHOULD NOT	No change
Fixed DH certificate types	Unspecified	SHOULD NOT
Ephemeral FFDH	SHOULD NOT	MUST NOT
Static RSA	SHOULD NOT	MUST NOT

Table 6

7. IANA Considerations

This document requests IANA to mark the cipher suites from the "TLS Cipher Suites" registry [tls-registry], under “Transport Layer Security (TLS) Parameters” registry group, listed in Section 5.1, Section 5.2, Section 5.3, Section 5.4, and the certificate types from the "TLS ClientCertificateType Identifiers" registry listed in Section 5.5 as "D" in the "Recommended" column, see [I-D.ietf-tls-rfc8447bis].

For each registry entry in Section 5.1, Section 5.2, Section 5.3, Section 5.4, and Section 5.5, IANA is also requested to update the registry entry’s Reference column to refer to this document.

8. Security Considerations

Non-ephemeral finite field DH cipher suites (TLS_DH_*), as well as ephemeral key reuse for finite field DH cipher suites, are prohibited due to the [Raccoon] attack. Both are already considered bad practice since they do not provide forward secrecy. However, Raccoon revealed that timing side channels in processing TLS premaster secrets may be exploited to reveal the encrypted premaster secret.

As for non-ephemeral elliptic curve DH cipher suites (TLS_ECDH_*), forgoing forward secrecy not only allows retroactive decryption in the event of key compromise but may also enable a broad category of attacks where the attacker exploits key reuse to repeatedly query a cryptographic secret.

This category includes, but is not necessarily limited to, the following examples:

1. Invalid curve attacks, where the attacker exploits key reuse to repeatedly query and eventually learn the key itself. These attacks have been shown to be practical against real-world TLS implementations [ICA].

2. Side channel attacks, where the attacker exploits key reuse and an additional side channel to learn a cryptographic secret. As one example of such attacks, refer to [MAY4].

3. Fault attacks, where the attacker exploits key reuse and incorrect calculations to learn a cryptographic secret. As one example of such attacks, see [PARIS256].

Such attacks are often implementation-dependent, including the above examples. However, these examples demonstrate that building a system that reuses keys and avoids this category of attacks is difficult in practice. In contrast, avoiding key reuse not only prevents decryption in the event of key compromise, but also precludes this category of attacks altogether. Therefore, this document discourages the reuse of elliptic curve DH public keys.

As for ephemeral finite field Diffie-Hellman in (D)TLS 1.2 (TLS_DHE_* and TLS_PSK_DHE_*), as explained above, clients have no practical way to support these cipher suites while ensuring they only negotiate security parameters that are acceptable to them. In (D)TLS 1.2, the server chooses the Diffie-Hellman group, and custom groups are prevalent. Therefore, once the client includes these cipher suites in its handshake and the server presents a custom group, the client cannot complete the handshake while ensuring security. Verifying the group structure is prohibitively expensive for the client. Using a safelist of known-good groups is also impractical, since server operators were encouraged to generate their own custom group. Further, there is no mechanism for the handshake to fall back to other parameters, that are acceptable to both the client and server.

9. Acknowledgments

This document includes many important contributions from Carrie Bartle, who wrote much of the prose, and presented it several times at the IETF TLS WG.

The document was inspired by discussions on the TLS WG mailing list and a suggestion by Filippo Valsorda following the release of the [Raccoon] attack. Thanks to Christopher A. Wood for writing up the initial draft of this document. Thanks also to Thomas Fossati, Sean Turner, Joe Salowey, Yaron Sheffer, Christian Buchgraber, John Preu Mattsson and Manuel Pgouri-Gonnard for comments and suggestions.

10. References

10.1. Normative References

[I-D.ietf-tls-rfc8446bis]
Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", Work in Progress, Internet-Draft, draft-ietf-tls-rfc8446bis-14, 13 September 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-rfc8446bis-14>.

[I-D.ietf-tls-rfc8447bis]
Salowey, J. A. and S. Turner, "IANA Registry Updates for TLS and DTLS", Work in Progress, Internet-Draft, draft-ietf-tls-rfc8447bis-15, 21 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-rfc8447bis-15>.

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

[RFC4162] Lee, H.J., Yoon, J.H., and J.I. Lee, "Addition of SEED Cipher Suites to Transport Layer Security (TLS)", RFC 4162, DOI 10.17487/RFC4162, August 2005, <https://www.rfc-editor.org/rfc/rfc4162>.

[RFC4279] Eronen, P., Ed. and H. Tschofenig, Ed., "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", RFC 4279, DOI 10.17487/RFC4279, December 2005, <https://www.rfc-editor.org/rfc/rfc4279>.

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

[RFC4785] Blumenthal, U. and P. Goel, "Pre-Shared Key (PSK) Ciphersuites with NULL Encryption for Transport Layer Security (TLS)", RFC 4785, DOI 10.17487/RFC4785, January 2007, <https://www.rfc-editor.org/rfc/rfc4785>.

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

[RFC5288] Salowey, J., Choudhury, A., and D. McGrew, "AES Galois Counter Mode (GCM) Cipher Suites for TLS", RFC 5288, DOI 10.17487/RFC5288, August 2008, <https://www.rfc-editor.org/rfc/rfc5288>.

[RFC5289] Rescorla, E., "TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)", RFC 5289, DOI 10.17487/RFC5289, August 2008, <https://www.rfc-editor.org/rfc/rfc5289>.

[RFC5469] Eronen, P., Ed., "DES and IDEA Cipher Suites for Transport Layer Security (TLS)", RFC 5469, DOI 10.17487/RFC5469, February 2009, <https://www.rfc-editor.org/rfc/rfc5469>.

[RFC5487] Badra, M., "Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode", RFC 5487, DOI 10.17487/RFC5487, March 2009, <https://www.rfc-editor.org/rfc/rfc5487>.

[RFC5932] Kato, A., Kanda, M., and S. Kanno, "Camellia Cipher Suites for TLS", RFC 5932, DOI 10.17487/RFC5932, June 2010, <https://www.rfc-editor.org/rfc/rfc5932>.

[RFC6209] Kim, W., Lee, J., Park, J., and D. Kwon, "Addition of the ARIA Cipher Suites to Transport Layer Security (TLS)", RFC 6209, DOI 10.17487/RFC6209, April 2011, <https://www.rfc-editor.org/rfc/rfc6209>.

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

[RFC6367] Kanno, S. and M. Kanda, "Addition of the Camellia Cipher Suites to Transport Layer Security (TLS)", RFC 6367, DOI 10.17487/RFC6367, September 2011, <https://www.rfc-editor.org/rfc/rfc6367>.

[RFC6655] McGrew, D. and D. Bailey, "AES-CCM Cipher Suites for Transport Layer Security (TLS)", RFC 6655, DOI 10.17487/RFC6655, July 2012, <https://www.rfc-editor.org/rfc/rfc6655>.

[RFC7905] Langley, A., Chang, W., Mavrogiannopoulos, N., Strombergson, J., and S. Josefsson, "ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS)", RFC 7905, DOI 10.17487/RFC7905, June 2016, <https://www.rfc-editor.org/rfc/rfc7905>.

[RFC7919] Gillmor, D., "Negotiated Finite Field Diffie-Hellman Ephemeral Parameters for Transport Layer Security (TLS)", RFC 7919, DOI 10.17487/RFC7919, August 2016, <https://www.rfc-editor.org/rfc/rfc7919>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8422] Nir, Y., Josefsson, S., and M. Pegourie-Gonnard, "Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS) Versions 1.2 and Earlier", RFC 8422, DOI 10.17487/RFC8422, August 2018, <https://www.rfc-editor.org/rfc/rfc8422>.

[RFC8996] Moriarty, K. and S. Farrell, "Deprecating TLS 1.0 and TLS 1.1", BCP 195, RFC 8996, DOI 10.17487/RFC8996, March 2021, <https://www.rfc-editor.org/rfc/rfc8996>.

[RFC9147] Rescorla, E., Tschofenig, H., and N. Modadugu, "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3", RFC 9147, DOI 10.17487/RFC9147, April 2022, <https://www.rfc-editor.org/rfc/rfc9147>.

[RFC9325] Sheffer, Y., Saint-Andre, P., and T. Fossati, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 9325, DOI 10.17487/RFC9325, November 2022, <https://www.rfc-editor.org/rfc/rfc9325>.

10.2. Informative References

[BLEI] Bleichenbacher, D., "Chosen Ciphertext Attacks against Protocols Based on RSA Encryption Standard PKCS #1", Advances in Cryptology -- CRYPTO’98, LNCS vol. 1462, pages: 1-12 , 1998.

[DLOG795] Boudot, F., Gaudry, P., Guillevic, A., Heninger, N., Thom, E., and P. Zimmermann, "Comparing the difficulty of factorization and discrete logarithm: a 240-digit experiment", 17 August 2020, <https://eprint.iacr.org/2020/697>.

[DROWN] Aviram, N., Schinzel, S., Somorovsky, J., Heninger, N., Dankel, M., Steube, J., Valenta, L., Adrian, D., Halderman, J. A., Dukhovni, V., Ksper, E., Cohney, S., Engels, S., Paar, C., and Y. Shavitt, "DROWN: Breaking TLS using SSLv2", August 2016, <https://drownattack.com/drown-attack-paper.pdf>.

[ICA] Jager, T., Schwenk, J., and J. Somorovsky, "Practical invalid curve attacks on TLS-ECDH", 21 September 2015, <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.704.7932&rep=rep1&type=pdf>.

[MAY4] Genkin, D., Valenta, L., and Y. Yarom, "May the fourth be with you: A microarchitectural side channel attack on several real-world applications of curve25519", n.d., <https://dl.acm.org/doi/pdf/10.1145/3133956.3134029>.

[NEW-BLEI] Meyer, C., Somorovsky, J., Weiss, E., Schwenk, J., Schinzel, S., and E. Tews, "Revisiting SSL/TLS Implementations: New Bleichenbacher Side Channels and Attacks", August 2014, <https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-meyer.pdf>.

[PARIS256] Devlin, S. and F. Valsorda, "The PARIS256 Attack", n.d., <https://i.blackhat.com/us-18/Wed-August-8/us-18-Valsorda-Squeezing-A-Key-Through-A-Carry-Bit-wp.pdf>.

[Raccoon] Merget, R., Brinkmann, M., Aviram, N., Somorovsky, J., Mittmann, J., and J. Schwenk, "Raccoon Attack: Finding and Exploiting Most-Significant-Bit-Oracles in TLS-DH(E)", 9 September 2020, <https://raccoon-attack.com/RacoonAttack.pdf>.

[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, <https://www.rfc-editor.org/rfc/rfc4492>.

[ROBOT] Boeck, H., Somorovsky, J., and C. Young, "Return Of Bleichenbacher’s Oracle Threat (ROBOT)", 27th USENIX Security Symposium , 2018.

[subgroups]
Valenta, L., Adrian, D., Sanso, A., Cohney, S., Fried, J., Hastings, M., Halderman, J. A., and N. Heninger, "Measuring small subgroup attacks against Diffie-Hellman", 15 October 2016, <https://eprint.iacr.org/2016/995/20161017:193515>.

[tls-registry]
"Transport Layer Security (TLS) Parameters", n.d., <https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml>.

[weak-dh] Adrian, D., Bhargavan, K., Durumeric, Z., Gaudry, P., Green, M., Halderman, J. A., Heninger, N., Springall, D., Thom, E., Valenta, L., VanderSloot, B., Wustrow, E., Zanella-Bguelin, S., and P. Zimmermann, "Weak Diffie-Hellman and the Logjam Attack", October 2015, <https://weakdh.org/>.

[XPROT] Jager, T., Schwenk, J., and J. Somorovsky, "On the Security of TLS 1.3 and QUIC Against Weaknesses in PKCS#1 v1.5 Encryption", Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security , 2015.

Author’s Address

Nimrod Aviram
Email: nimrod.aviram@gmail.com