Transport Layer Security
Internet-Draft
Intended status: Standards Track
Expires: 13 April 2026

D. Benjamin
Google LLC
A. Popov
Microsoft Corp.

10 October 2025

Legacy RSASSA-PKCS1-v1_5 codepoints for TLS 1.3
draft-ietf-tls-tls13-pkcs1-06

Abstract

This document allocates code points for the use of RSASSA-PKCS1-v1_5 with client certificates in TLS 1.3. This removes an obstacle for some deployments to migrate to TLS 1.3.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://tlswg.github.io/tls13-pkcs1/draft-ietf-tls-tls13-pkcs1.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-tls-tls13-pkcs1/.

Discussion of this document takes place on the Transport Layer Security Working Group mailing list (mailto:tls@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/tls/. Subscribe at https://www.ietf.org/mailman/listinfo/tls/.

Source for this draft and an issue tracker can be found at https://github.com/tlswg/tls13-pkcs1.

Status of This Memo

This Internet-Draft will expire on 13 April 2026.

Table of Contents

1. Introduction 2
2. Conventions and Definitions 3
3. PKCS#1 v1.5 SignatureScheme Types 3
4. Security Considerations 4
5. IANA Considerations 4
6. References 5
6.1. Normative References 5
6.2. Informative References 6
Acknowledgements 6
Authors' Addresses 6

1. Introduction

TLS 1.3 [RFC8446] removed support for RSASSA-PKCS1-v1_5 [RFC8017] in CertificateVerify messages in favor of RSASSA-PSS. While RSASSA-PSS is a long-established signature algorithm, some legacy hardware cryptographic devices lack support for it. While uncommon in TLS servers, these devices are sometimes used by TLS clients for client certificates.

For example, Trusted Platform Modules (TPMs) are ubiquitous hardware cryptographic devices that are often used to protect TLS client certificate private keys. However, a large number of TPMs are unable to produce RSASSA-PSS signatures compatible with TLS 1.3. TPM specifications prior to 2.0 did not define RSASSA-PSS support (see Section 5.8.1 of [TPM12]). TPM 2.0 includes RSASSA-PSS, but only those TPM 2.0 devices compatible with FIPS 186-4 can be relied upon to use the salt length matching the digest length, as required for compatibility with TLS 1.3 (see Appendix B.7 of [TPM2]).

TLS connections that rely on such devices cannot migrate to TLS 1.3. Staying on TLS 1.2 leaks the client certificate to network attackers [PRIVACY] and additionally prevents such deployments from protecting traffic against retroactive decryption by an attacker with a quantum computer [I-D.ietf-tls-hybrid-design].

Additionally, TLS negotiates protocol versions before client certificates. Clients send ClientHellos without knowing whether the server will request to authenticate with legacy keys. Conversely, servers respond with a TLS version and CertificateRequest without knowing if the client will then respond with a legacy key. If the client and server, respectively, offer and negotiate TLS 1.3, the connection will fail due to the legacy key, when it previously succeeded at TLS 1.2.

To recover from this failure, one side must globally disable TLS 1.3 or the client must implement an external fallback. Disabling TLS 1.3 impacts connections that would otherwise be unaffected by this issue, while external fallbacks break TLS's security analysis and may introduce vulnerabilities [POODLE].

This document allocates code points to use these legacy keys with client certificates in TLS 1.3. This reduces the pressure on implementations to select one of these problematic mitigations and unblocks TLS 1.3 deployment.

2. Conventions and Definitions

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.

3. PKCS#1 v1.5 SignatureScheme Types

The following SignatureScheme values are defined for use with TLS 1.3.

```
enum {
    rsa_pkcs1_sha256_legacy(0x0420),
    rsa_pkcs1_sha384_legacy(0x0520),
    rsa_pkcs1_sha512_legacy(0x0620),
} SignatureScheme;
```

The above code points indicate a signature algorithm using RSASSA-PKCS1-v1_5 [RFC8017] with the corresponding hash algorithm as defined in [SHS]. They are only defined for signatures in the client

CertificateVerify message and are not defined for use in other contexts. In particular, servers intending to advertise support for RSASSA-PKCS1-v1_5 signatures in the certificates themselves should use the rsa_pkcs1_* constants defined in [RFC8446].

Clients MUST NOT advertise these values in the signature_algorithms extension of the ClientHello. They MUST NOT accept these values in the server CertificateVerify message.

Servers that wish to support clients authenticating with legacy RSASSA-PKCS1-v1_5-only keys MAY send these values in the signature_algorithms extension of the CertificateRequest message and accept them in the client CertificateVerify message. Servers MUST NOT accept these code points if not offered in the CertificateRequest message.

Clients with such legacy keys MAY negotiate the use of these signature algorithms if offered by the server. Clients SHOULD NOT negotiate them with keys that support RSASSA-PSS, though this may not be practical to determine in all applications. For example, attempting to test a key for support might display a message to the user or have other side effects.

TLS implementations SHOULD disable these code points by default. See Section 4.

4. Security Considerations

The considerations in Section 1 do not apply to server keys, so these new code points are forbidden for use with server certificates. RSASSA-PSS continues to be required for TLS 1.3 servers using RSA keys. This minimizes the impact to only those cases necessary to unblock TLS 1.3 deployment.

When implemented incorrectly, RSASSA-PKCS1-v1_5 admits signature forgeries [MFSA201473]. Implementations producing or verifying signatures with these algorithms MUST implement RSASSA-PKCS1-v1_5 as specified in section 8.2 of [RFC8017]. In particular, clients MUST include the mandatory NULL parameter in the DigestInfo structure and produce a valid DER [X690] encoding. Servers MUST reject signatures which do not meet these requirements.

5. IANA Considerations

IANA is requested to create the following entries in the TLS SignatureScheme registry, defined in [RFC8446]. The "Recommended" column should be set to "N", and the "Reference" column should be set to this document.

Value	Description
0x0420	rsa_pkcs1_sha256_legacy
0x0520	rsa_pkcs1_sha384_legacy
0x0620	rsa_pkcs1_sha512_legacy

Table 1

6. References

6.1. 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, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8017] Moriarty, K., Ed., Kaliski, B., Jonsson, J., and A. Rusch, "PKCS #1: RSA Cryptography Specifications Version 2.2", RFC 8017, DOI 10.17487/RFC8017, November 2016, <https://www.rfc-editor.org/rfc/rfc8017>.

[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>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

[SHS] "Secure hash standard", National Institute of Standards and Technology (U.S.), DOI 10.6028/nist.fips.180-4, 2015, <https://doi.org/10.6028/nist.fips.180-4>.

[TPM12] Trusted Computing Group, "TPM Main Specification Level 2 Version 1.2, Revision 116, Part 2 - Structures of the TPM", 1 March 2011, <https://trustedcomputinggroup.org/wp-content/uploads/TPM-Main-Part-2-TPM-Structures_v1.2_rev116_01032011.pdf>.

[TPM2] Trusted Computing Group, "Trusted Platform Module Library Specification, Family 2.0, Level 00, Revision 01.59, Part 1: Architecture", 8 November 2019, <https://trustedcomputinggroup.org/wp-content/uploads/TCG_TPM2_r1p59_Part1_Architecture_pub.pdf>.

[X690] ITU-T, "Information technology - ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ISO/IEC 8825-1:2002, 2002.

6.2. Informative References

[I-D.ietf-tls-hybrid-design]
Stebila, D., Fluhrer, S., and S. Gueron, "Hybrid key exchange in TLS 1.3", Work in Progress, Internet-Draft, draft-ietf-tls-hybrid-design-16, 7 September 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-hybrid-design-16>.

[MFSA201473]
Delignat-Lavaud, A., "RSA Signature Forgery in NSS", 23 September 2014, <https://www.mozilla.org/en-US/security/advisories/mfsa2014-73/>.

[POODLE] Moeller, B., "This POODLE bites: exploiting the SSL 3.0 fallback", 14 October 2014, <https://security.googleblog.com/2014/10/this-poodle-bites-exploiting-ssl-30.html>.

[PRIVACY] Wachs, M., Scheitle, Q., and G. Carle, "Push away your privacy: Precise user tracking based on TLS client certificate authentication", IEEE, 2017 Network Traffic Measurement and Analysis Conference (TMA) pp. 1-9, DOI 10.23919/tma.2017.8002897, June 2017, <https://doi.org/10.23919/tma.2017.8002897>.

Acknowledgements

Thanks to Rifaat Shekh-Yusef, Martin Thomson, and Paul Wouters for providing feedback on this document.

Authors' Addresses

David Benjamin
Google LLC
Email: davidben@google.com

Andrei Popov
Microsoft Corp.
Email: andreipo@microsoft.com