Network Working Group
Internet-Draft
Updates: 6376 (if approved)
Intended status: Standards Track
Expires: February 1, 2018

S. Kitterman
Kitterman Technical Services

July 31, 2017

Cryptographic Algorithm and Key Usage Update to DKIM
draft-ietf-dcrup-dkim-usage-03

Abstract

The cryptographic algorithm and key size requirements included when DKIM was designed in the last decade are functionally obsolete and in need of immediate revision. This document updates DKIM requirements to those minimaly suitable for operation with currently specified algorithms.

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 February 1, 2018.

Copyright Notice

Table of Contents

1. Discussion Venue

RFC EDITOR: Please remove this section before publication.

Discussion about this draft is directed to the dcrup@ietf.org [1] mailing list.

2. Introduction

DKIM [RFC6376] signs e-mail messages, by creating hashes of the message headers and content and signing the header hash with a digital signature. Message recipients fetch the signature verification key from the DNS where it is stored in a TXT record.

The defining documents specify a single signing algorithm, RSA [RFC8017], and recommends key sizes of 1024 to 2048 bits (but require verification of 512 bit keys). As discussed in US-CERT VU#268267 [VULNOTE], the operational community has recognized that shorter keys compromise the effectiveness of DKIM. While 1024 bit signatures are common, stronger signatures are not. Widely used DNS configuration software places a practical limit on key sizes, because the software only handles a single 256 octet string in a TXT record, and RSA keys significantly longer than 1024 bits don't fit in 256 octets.

Due to the recognized weakness of the sha1 hash algorithm, see [RFC6194], and the wide availability of the sha256 hash algorithm (it has been a required part of DKIM [RFC6376] since it was originally standardized in 2007, the sha1 hash algorithm is removed from the protocol. This is being done now to allow the operational community time to fully shift to sha256 in advance of any sha1 related crisis.

3. Conventions Used in This Document

The capitalized 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 [RFC2119].

4. DKIM Signing and Verification Algorithms

This section replaces [RFC6376] Section 3.3 in its entirety.

Generally, DKIM supports multiple digital signature algorithms. One algorithm, rsa-sha256, is currenlty defined. Signers MUST implement and sign using rsa-sha256. Verifiers MUST implement and verify using rsa-sha256.

4.1. The rsa-sha256 Signing Algorithm

The rsa-sha256 Signing Algorithm computes a message hash as described in [RFC6376], Section 3.7 using SHA-256 [FIPS-180-3-2008] as the hash-alg. That hash is then signed by the Signer using the RSA algorithm (defined in PKCS#1 version 1.5 [RFC8017]) as the crypt-alg and the Signer's private key. The hash MUST NOT be truncated or converted into any form other than the native binary form before being signed. The signing algorithm SHOULD use a public exponent of 65537.

4.2. Key Sizes

Selecting appropriate key sizes is a trade-off between cost, performance, and risk. Since short RSA keys more easily succumb to off-line attacks, Signers MUST use RSA keys of at least 1024 bits for all keys. Verifiers MUST be able to validate signatures with keys ranging from 1024 bits to 4096 bits, and they MAY be able to validate signatures with larger keys. Verifier policies can use the length of the signing key as one metric for determining whether a signature is acceptable.

Factors that should influence the key size choice include the following:

- The practical constraint that large (e.g., 4096-bit) keys might not fit within a 512-byte DNS UDP response packet
- The security constraint that keys smaller than 2048 bits may be subject to off-line attacks
- Larger keys impose higher CPU costs to verify and sign email
- Keys can be replaced on a regular basis; thus, their lifetime can be relatively short
- The security goals of DKIM [RFC6376] are modest compared to typical goals of other systems that employ digital signatures

See [RFC3766] for further discussion on selecting key sizes.

4.3. Other Algorithms

The rsa-sha1 was formerly used by DKIM [RFC6376]. Signers MUST NOT sign with rsa-sha1 and verifiers MUST NOT verify using rsa-sha1.

Other algorithms will be defined in the future. Verifiers MUST ignore any signatures using algorithms that they do not implement.

5. The DKIM-Signature Header Field

This section updates the a= tag in [RFC6376] Section 3.5.

The text description of the tag is now:

a= The algorithm used to generate the signature (plain-text; REQUIRED). Verifiers MUST support "rsa-sha256"; Signers MUST sign using "rsa-sha256". See [RFC6376] Section 3.3 (as updated by this document) for a description of the algorithms.

The following ABNF element is updated:

```
ABNF:

sig-a-tag-h     = "sha256" / x-sig-a-tag-h
```

6. Key Management and Representation

This section updates the h= tag in [RFC6376] Section 3.6.1.

The following ABNF element is updated:

ABNF:

```
key-h-tag-alg   = "sha256" / x-key-h-tag-alg
```

7. Security Considerations

This document does not change the Security Considerations of [RFC6376]. It reduces the risk of signature compromise due to weak cryptography. The SHA-1 risks discussed in [RFC6194] Section 3 are resolved due to the removal of rsa-sha1 from DKIM.

8. IANA Considerations

IANA is requested to update the "sha1" registration in the "DKIM Hash Algorithms" as follows:

TYPE	REFERENCE	STATUS
sha1	(this document)	obsolete

Table 1: DKIM Hash Algorithms Changed Value

9. References

9.1. 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>.

[RFC3766] Orman, H. and P. Hoffman, "Determining Strengths For Public Keys Used For Exchanging Symmetric Keys", BCP 86, RFC 3766, DOI 10.17487/RFC3766, April 2004, <http://www.rfc-editor.org/info/rfc3766>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <http://www.rfc-editor.org/info/rfc6376>.

[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, <http://www.rfc-editor.org/info/rfc8017>.

9.2. Informative References

[RFC6194] Polk, T., Chen, L., Turner, S., and P. Hoffman, "Security Considerations for the SHA-0 and SHA-1 Message-Digest Algorithms", RFC 6194, DOI 10.17487/RFC6194, March 2011, <http://www.rfc-editor.org/info/rfc6194>.

[VULNOTE] US-CERT, "Vulnerability Note VU#268267, DomainKeys Identified Mail (DKIM) Verifiers may inappropriately convey message trust", October 2012.

9.3. URIs

[1] mailto:dcrup@ietf.org

Appendix A. Acknowledgements

The author wishes to acknowledge the following for their review and comment on this proposal: Kurt Andersen, Murray S. Kucherawy, Martin Thomson, John Levine, Russ Housley, and Jim Fenton.

Thanks to John Levine his DCRUP work that was the source for much of the introductory material in this draft.

Author's Address

Scott Kitterman
Kitterman Technical Services
3611 Scheel Dr
Ellicott City, MD 21042

Phone: +1 301 325-5475
Email: scott@kitterman.com