DCRUP
Internet-Draft
Updates: 6376 (if approved)
Intended status: Standards Track
Expires: December 4, 2017

S. Rose
NIST

June 2, 2017

Defining Elliptic Curve Cryptography Algorithms for use with DKIM
draft-ietf-dcrup-dkim-ecc-00

Abstract

DomainKeys Identified Mail (DKIM) uses digital signature to associate a message with a given sending domain. Currently, there is only one cryptography algorithm defined for use with DKIM (RSA). This document defines four new elliptic curve cryptography algorithms for use with DKIM. This will allow for algorithm agility if a weakness is found in RSA, and allows for smaller key length to provide the same digital signature strength.

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 December 4, 2017.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Requirements Language 3
2. Defining New ECC algorithms for Use with DKIM 3
3. Changes to ABNF Definitions of DKIM Keys and Signatures . . . 3
3.1. Changes to DKIM Key Definition 3
3.2. Changes to DKIM Signature Definition 4
4. Sender Considerations 5
5. Receiver Considerations 5
6. Security Considerations 5
7. IANA Considerations 6
8. References 6
8.1. Normative References 6
8.2. Informative References 7
Author's Address 7

1. Introduction

DomainKeys Identified Mail (DKIM)[RFC6376] uses digital signatures to associate a sending domain with a given message. Each DKIM signed email message as a digital signature in its header, that can be validated by a receiver by obtaining the appropriate public key stored in the DNS. Currently, DKIM has only one cryptographic algorithm defined for use (RSA) and two digital signature algorithms (RSA/SHA-1 and RSA/SHA-256). In the past, 1024-bit RSA keys were common, equating to (roughly) a security key strength of 80 bits [NIST.800-57.2016]. Today, a minimum of 112 bits is recommended, which equates to 2048 bit RSA keys.

The public portion of 2048 bit RSA keys are still small enough to fit into a DNS TXT RR without issues in performance. The encoded public key is too large to fit into the maximum allowed characters in a single string, but a DNS TXT RR allows for multiple strings, so the key can be broken into "chunks" to allow it to be served. However, some code components may not correctly handle TXT RRs with multiple strings which will result in errors in validation.

Elliptic Curve Cryptography (ECC) has shown to have the same

(roughly) equivalent key strength with smaller sizes. A 224 to 255 bit ECDSA key has (roughly) the same key strength as a 2048 bit RSA key (112 bits of strength). This means smaller keys can be used to achieve the same DKIM security strength, as well as being easier to manage in the DNS.

Having additional digital signature algorithms defined for use with DKIM also permits algorithm agility. If a weakness is discovered in one digital signature algorithm, email senders can quickly migrate to another algorithm without waiting for a standards action and subsequent software update.

This document defines a ECDSA as a new algorithms for DKIM. This document also defines a new hash algorithm for use with DKIM signatures. This document updates the IANA registry with new values for the algorithms. This document does not change the DKIM key or signature formats, but only defines new algorithm values using those formats.

1.1. Requirements Language

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].

2. Defining New ECC algorithms for Use with DKIM

This document defines a new digital signature algorithm for use with DKIM:

algorithm	mnemonic
ECDSA P-256	ecdsa256

The SHA-512 hash algorithm is also now defined for use with DKIM using the mnemonic 'sha512' for the "h=" DKIM key tag and "a=" sig-a-tag-h DKIM signature tag (see below).

For ECDSA, the SHA-1 hash algorithm MUST NOT be used. Both ECDSA and RSA MAY be used with SHA-512.

3. Changes to ABNF Definitions of DKIM Keys and Signatures

The original definition of DKIM signatues and keys are defined in [RFC6376]. The following are changes to the definition to include the new digital signature algorithm and secure hash algorithm.

3.1. Changes to DKIM Key Definition

The original definition of the textual representation of DKIM keys is found in section 3.6.1 of [RFC6376]. The only changes to the definition is below. The entire key:tag definition is included for clarity. All other tags:value pairs are unchanged. References to

the definitions below have also been updated to reflect the current state of the art.

h= Acceptable hash algorithms (plain-text; OPTIONAL, defaults to "sha256"). A colon-separated list of hash algorithms that might be used. Unrecognized algorithms MUST be ignored. Refer to [RFC6376]Section 3.3 for a discussion of the hash algorithms implemented by Signers and Verifiers. The set of algorithms listed in this tag in each record is an operational choice made by the Signer.

ABNF:

key-h-tag = %x68 [FWS] "=" [FWS] key-h-tag-alg *([FWS] ":" [FWS] key-h-tag-alg) key-h-tag-alg = "sha1" / "sha256" / "sha512" / x-key-h-tag-alg x-key-h-tag-alg = hyphenated-word ; for future extension

k= Key type (plain-text; MANDATORY). Signers and Verifiers MUST support the "rsa" key type. The "rsa" key type indicates that an ASN.1 DER-encoded [UTI.X680.2002] RSAPublicKey (see [RFC8017], Sections 3.1 and A.1.1) is being used in the "p=" tag. The "ecdsa256" key type indicates an ASN.1 DER-encoded [UTI.X680.2002] PublicKey (see [RFC5480], Section 2.2) is being used in the "p=" tag. (Note: the "p=" tag further encodes the value using the base64 algorithm.) Unrecognized key types MUST be ignored.

ABNF:

key-k-tag = %x76 [FWS] "=" [FWS] key-k-tag-type key-k-tag-type = "rsa" / "ecdsa256" / x-key-k-tag-type x-key-k-tag-type = hyphenated-word ; for future extension

3.2. Changes to DKIM Signature Definition

The original definition of the textual representation of DKIM signatures is found in section 3.5 of [RFC6376]. The only changes to the definition is below. The entire key:tag definition is included for clarity. All other tags:value pairs are unchanged. References to the definitions below have also been updated to reflect the current state of the art.

a= The algorithm used to generate the signature (plain-text; REQUIRED). Verifiers MUST support "rsa-sha1" and "rsa-sha256" and SHOULD support "ecdsa256-sha256"; Signers MUST NOT use "sha1" with "ecdsa256". See [RFC6376] Section 3.3 for a description of RSA and [FIPS.186-4.2013] Section 6 for a brief description of ECDSA.

ABNF:

sig-a-tag = %x61 [FWS] "=" [FWS] sig-a-tag-alg sig-a-tag-alg = sig-a-tag-k "-" sig-a-tag-h sig-a-tag-k = "rsa" / "ecdsa256" / x-sig-a-tag-k sig-a-tag-h = "sha1" / "sha256" / "sha512" / x-sig-a-tag-h x-sig-a-tag-k = ALPHA *(ALPHA / DIGIT) ; for later extension x-sig-a-tag-h = ALPHA *(ALPHA / DIGIT) ; for later extension

4. Sender Considerations

New algorithms for an established protocols take some time to gain wide deployment. There will be a period of time where new algorithms are in operation side by side with older algorithms. There will also be a sizable percentage of DKIM validators that will not understand new algorithms until they are upgraded. This will lead to a period of time where multiple DKIM signature algorithms are in use for a sender. Email administrators MAY want to also sign with RSA/SHA-1 or RSA/SHA-256 for a period of time. This period of time is difficult to measure, but DMARC [RFC7960] aggregate reports could provide a view on DKIM validation rates by receivers.

5. Receiver Considerations

These requirements are for DKIM verifiers (as defined it [RFC6376]). These entities would be the consumers of any end-to-end email security policy and would be the entity responsible for validating DKIM signatures.

DKIM verifiers claiming conformance to this document MUST implement all of the above cryptographic algorithms and SHOULD implement the SHA-512 hash algorithm.

This document does NOT change the behavior of the core DKIM specification in that verifiers MUST ignore unknown algorithms in DKIM signatures.

6. Security Considerations

This document defines the use of new elliptic curve cryptographic algorithms for use with DomainKey Identified Mail (DKIM). This document is not a discussion of the relative strengths or weaknesses of these algorithms, but only defines their use.

There is a risk for mail receivers that do not understand or implement the new algorithms. Attackers could modify or spoof messages from sending zones using one of the newly defined algorithms and it would not be detectable as an attack by ECC-ignorant

receivers. Likewise, ECC-ignorant receivers may mark valid DKIM signed email messages as invalid due to unknown algorithms.

7. IANA Considerations

This draft defines the use of a new algorithm for DKIM. This draft updates the "DKIM Key Tag" registry to include the following new value:

algorithm	mnemonic	Reference
ECDSA P-256	ecdsa256	This document

The current DKIM Key Tag registry is located at https://www.iana.org/assignments/dkim-parameters/dkim-parameters.xhtml#dkim-parameters-6

This draft also defines a new hash algorithm for use with DKIM. This draft updates the "DKIM Hash Algorithms" registry to include the following new entry:

algorithm	mnemonic	Reference
SHA-512	sha512	This document

The current DKIM Hash Algorithm registry is located at https://www.iana.org/assignments/dkim-parameters/dkim-parameters.xhtml#dkim-parameters-7

8. References

8.1. 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>.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, DOI 10.17487/RFC5480, March 2009, <http://www.rfc-editor.org/info/rfc5480>.

[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>.

[UTI.X680.2002]
"ITU-T Recommendation X.680 (2002) | ISO/IEC 8825-1:2002, Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU X680, 2002.

8.2. Informative References

[FIPS.186-4.2013]
National Institute of Standards and Technology, "Digital Signature Standard", FIPS PUB 186-4, July 2013, <http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf>.

[NIST.800-57.2016]
National Institute of Standards and Technology, "Recommendations for Key Management Part 1: General", NIST 800-57, January 2016.

[RFC7960] Martin, F., Ed., Lear, E., Ed., Draegen. Ed., T., Zwicky, E., Ed., and K. Andersen, Ed., "Interoperability Issues between Domain-based Message Authentication, Reporting, and Conformance (DMARC) and Indirect Email Flows", RFC 7960, DOI 10.17487/RFC7960, September 2016, <http://www.rfc-editor.org/info/rfc7960>.

Author's Address

Scott Rose
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
USA

Phone: +1 301-975-8439
Email: scott.rose@nist.gov