

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 19, 2017

M. Kucherawy
November 15, 2016

Including Recipients in DKIM Signatures
draft-kucherawy-dkim-rcpts-01

Abstract

The DomainKeys Identified Mail (DKIM) protocol applies a domain-level cryptographic signature to an e-mail message. DKIM only guarantees authenticity of the message content and does not consider the message envelope. This allows for replay attacks by recycling a signed message with an arbitrary new set of recipients.

This document presents a protocol extension that can include original envelope information in the signature data, so that an altered that information renders the signature invalid.

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 May 19, 2017.

Copyright Notice

Copyright (c) 2016 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

Internet-Draft

DKIM Canonicalized Recipients

November 2016

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
2.	Definitions	3
3.	'nr' Tag Definition	3
4.	Implementation	4
4.1.	Signers	4
4.2.	Verifiers	5
5.	Compatibility with Current Infrastructure	5
6.	IANA Considerations	6
7.	Privacy Considerations	6
8.	Security Considerations	6
9.	Implementation Status	7
10.	Change Log	7
10.1.	01	7
10.2.	00	7
11.	References	7
11.1.	Normative References	7
11.2.	Informative References	8
Appendix A.	Acknowledgments	8
Author's Address	8

[1.](#) Introduction

DKIM [[RFC6376](#)] defines a cryptographic signature, placed in a header field consisting of a series of tags and values. The values include signed hashes of some of the header fields and part or all of the body of a message. The signature contains a domain name that is responsible for the signature and thus takes some responsibility for the presence of the message in the email stream.

The signature is valid if the hashes in the signature match the corresponding hashes of the message at validation time, the signature is validated by a public key retrieved from that responsible domain's DNS, and it is before the expiration time in the signature header field (if set).

There have been recent incidents of a replay attack, where a message

of undesirable content (spam, malware, phishing, etc.) is sent by a bad actor to itself through an email service, which dutifully signs it. This message now bears the digital signature of the signing agent's domain, which means in many cases that the signing agent's reputation will be weighed by a receiver when assessing the likely

safety of the message. The bad actor is then free to re-send that message to any number of other recipients with that same signature, any number of times, by altering the set of recipients on the message (the "envelope" in terms of the Simple Mail Transfer Protocol (SMTP) [[RFC5321](#)]) and re-sending it. This was anticipated by [[RFC6376](#) [Section 8.6](#)].

Obviously a signing agent would be well within its rights and own interests to decline to sign something that looks like it might be unwanted content, but such measures are not fool-proof. What is needed, then, is a way to thwart these sorts of replay attacks.

The proposal presented here is to include in the signature data the original recipient the message. A verifier could thereby confirm that the envelope recipient matches the envelope recipient that was used on the message when signed, and take defensive measures when a mismatch is identified.

For various operational reasons related to SMTP, covered in [Section 5](#), this extension cannot reliably accommodate messages with multiple envelope recipients, and so use of this extension with a message bearing multiple envelope recipients is undefined.

[2](#). Definitions

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 [[RFC2119](#)].

Syntax descriptions use Augmented BNF (ABNF) [[RFC5234](#)]. The definition of the "FWS" ABNF token is taken from [[RFC6376](#) [Section 2.8](#)]. The definition of the "base64string" token is taken from [[RFC6376](#) [Section 2.10](#)].

A full description of the email ecosystem can be found in [[RFC5598](#)].

The "envelope recipient" is the recipient identified in an SMTP [\[RFC5321\]](#) RCPT TO command.

[3.](#) 'nr' Tag Definition

The following DKIM tags (see [\[RFC6376\] Section 3.5](#)) are introduced:

rh= Recipient hash (base64; OPTIONAL).

ABNF:

sig-rh-tag = %x72.68 [FWS] "=" [FWS] base64string

Kucherawy

Expires May 19, 2017

[Page 3]

Internet-Draft

DKIM Canonicalized Recipients

November 2016

The output of the SHA hash of the envelope recipient, as described in [Section 4](#).

rs= Recipient salt (plain-text; OPTIONAL).

ABNF:

salt-chars = (ALPHA / DIGIT)

sig-rs-tag = %x72.73 [FWS] "=" [FWS] 1*8salt-chars

If present, this provides a salt that is prepended to the envelope recipient before hashing. Ignored if the "rh" tag is not also present.

[4.](#) Implementation

This section describes implementation of this extension in detail.

[4.1.](#) Signers

When producing the canonicalized header using this proposal, the signer takes the following steps:

1. Collect the SMTP recipient to be used for sending the message being signed.
2. Canonicalize the recipient string using NFKC per the string preparation framework described in [\[RFC7564\]](#).

3. OPTIONAL: Select a sequence of one to eight random alphanumeric ASCII characters. This is the encoding salt. Prepend this to the previous string, and add it to the DKIM-Signature header field being generated as the value of the "rs" tag.
4. Apply the same SHA transformation to the above string as is implied by the signing algorithm to be used in generating this signature. That is, apply SHA1 if the "a=" tag is "rsa-sha1" or SHA256 if the "a=" tag is "rsa-sha256". (See [[RFC6376](#)] for a definition of the "a=" tag.)
5. Add to the DKIM-Signature header field an "rh" tag whose value is the base64 encoding of the output of the SHA transformation in the previous step.
6. Continue with header canonicalization hashing, and DKIM-Signature header field construction as defined in [[RFC6376](#)].

[4.2](#). Verifiers

When analyzing the DKIM-Signature field on an arriving message that includes the "rh" tag defined in [Section 3](#), the verifier takes the following steps:

1. Collect the SMTP recipient to be used for sending the message being signed.
2. Canonicalize the recipient string using NFKC per the string preparation framework described in [[RFC7564](#)].
3. If an "rs" tag is present in the DKIM-Signature header field being evaluated, prepend its value to the string produced by the previous step.
4. Apply the same SHA transformation to the above string as is implied by the "a=" tag present in the DKIM-Signature header field being evaluated.
5. Apply base64 encoding to the output of the SHA transformation.

6. If the base64 encoding does not exactly match the value of the "rh" tag present in the DKIM-Signature header field being evaluated, report PERMFAIL for this signature and stop processing.
7. Continue with header canonicalization, hashing, and DKIM-Signature header field verification as defined in [[RFC6376](#)].

This has the effect of requiring the same recipient on the message at time of receipt (more precisely, at time of verification) as was there at the time of signing of the message. If that is not the case, the "rh" tag values produced at each end will fail to match. This effectively prevents the sort of attack described in [Section 1](#).

[5](#). Compatibility with Current Infrastructure

[RFC6376] [Section 3.5](#) requires verifiers to ignore tags they do not understand. Accordingly, the introduction of these tags by signers should have no negative impact on existing (correct) implementations.

The restriction on use for multiple-recipient messages is predicated on numerous operational issues, including:

- o Messages can be split anywhere along their handling path to direct the content along separate paths, such as when different recipients are handled by different mail exchanges;

- o Recording all recipients in this way would potentially expose hidden recipients (e.g., Bcc) to parties that would not otherwise be able to detect them;
- o A message indicating multiple recipients would fail to verify if some of those recipients were deferred by the receiving system for valid operational reasons such as recipient count limits or invalid recipients.

[6](#). IANA Considerations

IANA is requested to register the following in the "DKIM-Signature Tag Specifications" registry:

Type: rh

Reference: [this document]

Status: active

Type: rs

Reference: [this document]

Status: active

7. Privacy Considerations

The recipients of a message are not typically recorded anywhere in the message content itself and is instead a property of the SMTP "envelope" used to transport it that is discarded on delivery. This results in the ability to, among other things, do a "blind carbon copy" of a message that does not reveal one recipient to the others.

This proposal adds the full recipient address to the content presented for hashing and ultimate transmission of the message. It does not expose that content to receivers visibly, so there is not a direct leak of potentially private information. However, by attaching even an encoded form of the recipient allows an attacker to make an educated guess about who the recipient might be, repeat the algorithm described in [Section 4.2](#), and determine if the guess is correct.

8. Security Considerations

[Section 8 of \[RFC6376\]](#) enumerates known security issues with DKIM. In particular, [Section 8.6 of \[RFC6376\]](#) anticipated this attack.

The issues of compatibility discussed in [\[RFC6376\]](#) are unfortunately the ideal. It is possible or even likely that introducing a new DKIM tag that requires verifier participation for success will result in rejection of otherwise legitimate messages, the impact of which depends almost entirely on the sensitivity of the content thus rejected.

Apart from the privacy-specific discussion in [Section 7](#), and the

potential impact on current infrastructure discussed in [Section 5](#), no new security issues are introduced here.

[9.](#) Implementation Status

The next release of OpenDKIM will implement this proposal. OpenDKIM is in widespread use, including at very large installations, so use and utility of this extension can be easily observed.

[10.](#) Change Log

[10.1.](#) 01

- o Change "nr" to "rh" and "rs".

[10.2.](#) 00

- o Initial version.

[11.](#) References

[11.1.](#) 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>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), DOI 10.17487/RFC5234, January 2008, <<http://www.rfc-editor.org/info/rfc5234>>.
- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", [RFC 5321](#), DOI 10.17487/RFC5321, October 2008, <<http://www.rfc-editor.org/info/rfc5321>>.

"DomainKeys Identified Mail (DKIM) Signatures", STD 76, [RFC 6376](#), DOI 10.17487/RFC6376, September 2011, <<http://www.rfc-editor.org/info/rfc6376>>.

[RFC7564] Saint-Andre, P. and M. Blanchet, "PRECIS Framework: Preparation, Enforcement, and Comparison of Internationalized Strings in Application Protocols", [RFC 7564](#), DOI 10.17487/RFC7564, May 2015, <<http://www.rfc-editor.org/info/rfc7564>>.

11.2. Informative References

[RFC5598] Crocker, D., "Internet Mail Architecture", [RFC 5598](#), DOI 10.17487/RFC5598, July 2009, <<http://www.rfc-editor.org/info/rfc5598>>.

Appendix A. Acknowledgments

Valuable input to this proposal was provided by Michael Adkins, Peter Blair, Dave Crocker, Vladimir Dubrovin, Ned Freed, Steven Jones, John Levine, Scott Kitterman, Martijn Grooten, and Alexey Toptygin.

Author's Address

Murray S. Kucherawy
270 Upland Drive
San Francisco, CA 94127

Phone: +1 415 505 6296
Email: superuser@gmail.com