Internet Engineering Task Force (IETF)
Request for Comments: 6541
Category: Experimental
ISSN: 2070-1721

M. Kucherawy
Cloudmark, Inc.

February 2012

DomainKeys Identified Mail (DKIM) Authorized Third-Party Signatures

Abstract

This experimental specification proposes a modification to DomainKeys Identified Mail (DKIM) allowing advertisement of third-party signature authorizations that are to be interpreted as equivalent to a signature added by the administrative domain of the message's author.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6541.

Copyright Notice

Table of Contents

1. Introduction

[DKIM] defines a mechanism for transparent domain-level signing of messages for the purpose of declaring that a particular ADministrative Management Domain (ADMD) takes some responsibility for a message.

DKIM, however, deliberately makes no binding between the DNS domain of the Signer and any other identity found in the message. Despite this, there is an automatic human perception that an Author Domain Signature (one for which the RFC5322.From domain matches the DNS domain of the Signer) is more valuable or trustworthy than any other.

To enable a third party to apply DKIM signatures to messages, the DKIM specification suggests delegation to a third party of either subdomains or private keys, so that the third party can add DKIM signatures that appear to have been added by the Author ADMD. Absent is a protocol by which an Author ADMD can announce that messages bearing specific valid DKIM signatures on its mail, which are added by other ADMDs, are to be treated as if they were signed by the Author ADMD itself. This memo presents an experimental mechanism for doing so, called Authorized Third-Party Signatures (ATPS).

ATPS augments the semantics of DKIM by providing to the Verifier multiple identifiers rather than one. Specifically, it validates the identifier found in the DKIM signature, and then provides the RFC5322.From domain for evaluation.

This memo also registers, per [AUTHRES], the means to indicate to agents downstream of the Verifier that a third-party signature verification occurred.

2. Definitions

2.1. Key Words

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 [KEYWORDS].

2.2. Email Architecture Terminology

Readers are advised to be familiar with the material and terminology discussed in [MAIL] and [EMAIL-ARCH].

3. Roles and Scope

The context of this protocol involves the following roles:

- ADministrative Management Domains (ADMDs), whose DNS domain name(s) appear in the RFC5322.From field of a [MAIL] message;

- ATPS Signers, which apply [DKIM] signatures using their own domains, but on behalf of the message Author's ADMD; and

- the Verifier, who implements the signature validation procedures described in [DKIM].

An ADMD implements this protocol if it wishes to announce that a signature from any in a set of specified DNS domains is to be considered equivalent to one from the ADMD itself. For example, an ADMD might wish to delegate signing authority for its DNS domain to an approved messaging service provider without doing the work of key transfer described in Appendix B.1.1 of [DKIM]. An authorized ATPS

Signer makes a claim of this relationship via new tags in the DKIM signature, and the ADMD confirms this claim by publishing a specific TXT record in its DNS.

A Verifier implements this protocol if it wishes to ensure that a message bears one or more signatures from sources authorized to sign mail on behalf of the ADMD, and identify for special treatment mail that meets (or does not meet) that criterion. It will do so by treating the Signer's authorization on behalf of the Author's ADMD to mean that the Signer's signature is equivalent to one affixed by the Author's ADMD.

4. Queries and Replies

This section describes in detail the queries issued, the replies received, and how they should be interpreted and applied.

4.1. Hash Selection

The Author's ADMD will indicate authorization of a third party to sign its mail via the presence of a DNS TXT record that contains an encoding of the third party's DNS domain name. There are two supported methods for doing so -- one that involves a plain copy of the third party's DNS domain name, and one that involves an encoded version of the name. The encoding mechanism is provided so that any domain name can be added to the DNS in a fixed length, so that longer third-party domain names are not excluded from participation because of the overall length limit on a DNS query.

If selected, the encoding mechanism requires constructing a digest of the third party's DNS domain name. The Author ADMD MUST select a digest ("hash") method currently supported by DKIM (see Section 7.7 of [DKIM]), and this selection needs to be communicated to the ATPS Signer, as it is used in generation of the third-party signatures.

Where the encoding mechanism is not used, the ATPS Signer MUST use a hash name of "none".

The full DNS mechanism is specified in Section 4.3.

4.2. Extension to DKIM

[DKIM] signatures contain a "tag=value" sequence. This protocol will add additional tags called "atps" and "atpsh".

When the ATPS Signer generates a DKIM signature for another ADMD, it MUST put its own domain in the signature's "d" tag, and include an "atps" tag that has as its value the domain name of the ADMD on whose behalf it is signing.

The tag name that carries the name of the selected hash algorithm is "atpsh". This tag MUST also be included, as it is required as part of the algorithm that will be enacted by the Verifier.

The formal syntax definition, per [ABNF], is as follows:

```
dkim-atps-tag = %x61.74.70.73 *WSP "=" *WSP domain-name

dkim-atpsh-tag = %x61.74.70.73.68 *WSP "=" *WSP
                 ( "none" / key-h-tag-alg )
```

"domain-name" and "key-h-tag-alg" are defined in [DKIM]. Note that according to [DKIM], internationalized domain names are to be encoded as A-labels, as described in Section 2.3 of [IDNA].

The registration for these tags can be found in Section 8.

4.3. ATPS Query Details

When a [DKIM] signature including an "atps" tag is successfully verified, and is considered acceptable to the Verifier according to any local policy requirements (which are not discussed here or in [DKIM]), the Verifier compares the domain name in the value of that tag with the one found in the RFC5322.From field of the message. The match MUST be done in a case-insensitive manner.

If they do not match, the "atps" tag MUST be ignored.

If they do match, the Verifier issues a DNS TXT query, as specified below, looking for confirmation by the Author ADMD that the ATPS Signer is authorized by that ADMD to sign mail on its behalf. Where multiple DKIM signatures including valid "atps" tags are present, these queries MAY be done in any order or MAY be done in parallel.

Where the RFC5322.From field contains multiple addresses, this process SHOULD be applied if the "atps" tag's value matches any of the domains found in that field. These MAY be done in any order.

Note that the algorithm uses hashing, but this is not a security mechanism. See Section 9.2 for discussion.

The name for the query is constructed as follows:

1. Select the hash algorithm from the "atpsh" tag in the signature. If the hash algorithm specified does not appear in the list registered with IANA as one valid for use with DKIM (see Section 7.7 of [DKIM]), and is not the reserved name "none" as described above, abort the query.

2. Extract the value of the "d=" tag from the signature.

3. Convert any uppercase characters in that string to their lowercase equivalents.

4. If the selected hash algorithm is not "none", apply the following additional steps:

   A. Feed the resulting string to the selected hash algorithm.

   B. Convert the output of the hash to a string of printable ASCII characters by applying base32 encoding as defined in Section 6 of [BASE32]. The base32 encoding is used because its output is restricted to characters that are legal for use in labels in the DNS, and it is evaluated the same way in the DNS whether encoded using uppercase or lowercase characters.

5. Append the string "._atps."

6. Append the domain name found in the "atps" tag of the validated signature.

The query's formal syntax definition, per [ABNF], is as follows:

```
atps-query = ( 1*63BASE32 / domain-name )
             %x2e.5f.61.74.70.73.2e domain-name

BASE32 = ( ALPHA / %x32-37 )
```

The width limit of 63 on the base32 encoding is based on the maximum label limit as defined in Section 2.3.4 of [DNS].

See Appendix A for an example of a query construction.

4.4. ATPS Reply Details

In the descriptions below, the label NOERROR symbolizes DNS response code ("rcode") 0, and NXDOMAIN represents rcode 3. See Section 4.1.1 of [DNS] for further details.

At this time, only three possibilities need to be identified in this specification:

- An answer is returned (i.e., [DNS] reply code NOERROR with at least one answer) containing a valid ATPS reply. In this case, the protocol has been satisfied and the Verifier can conclude that the signing domain is authorized by the ADMD to sign its mail. Further queries SHOULD NOT be initiated.

- No answer is returned (i.e., [DNS] reply code NXDOMAIN, or NOERROR with no answers), or one or more answers have been returned as described above but none contain a valid ATPS reply. In this case, the Signer has not been authorized to act as a third-party Signer for this ADMD, and thus the Verifier MUST continue to the next query, if any.

- An error is returned (i.e., any other [DNS] reply code). It is no longer possible to determine whether or not this message satisfies the ADMD's list of authorized third-party Signers. The Verifier SHOULD stop processing and defer the message for later processing, such as requesting a temporary failure code from the Mail Transfer Agent (MTA).

If all queries are completed and return either NXDOMAIN or NOERROR with no answers, then the Signer was not authorized by the ADMD.

A valid ATPS reply consists of a sequence of tag=value pairs as described in Section 3.2 of [DKIM]. The following tags and values are currently supported in ATPS records:

d: Domain (plain-text; RECOMMENDED). This tag includes a plain-text copy of the DNS domain being authorized as an ATPS Signer. This is included to assist with collision detections; for example, if the base32 encoding of this name is not the same as the base32 portion of the query, or more simply if this name is not the same as that found in the "atps" tag, a hash collision could have occurred. Its use where no name hashing has occurred is redundant. The ABNF is as follows:

```
atps-d-tag = %x64 [FWS] "=" [FWS] domain-name
           ; FWS is defined in [DKIM]
```

v: Version (plain-text; REQUIRED). This tag indicates the version of the ATPS specification to which the record complies. The record MUST be ignored if the value is not "ATPS1". The ABNF is as follows:

```
atps-v-tag = %x76 [FWS] "=" [FWS] %x41.54.50.53.31
           ; FWS is defined in [DKIM]
```

5. Interpretation

For each DKIM signature that verifies (see Section 6 of [DKIM]), if a Verifier succeeds in confirming that the Author's ADMD authorized the ATPS Signer using this protocol, then the Verifier SHOULD evaluate the message as though it contained a valid signature from the Author's ADMD. It MAY also independently evaluate the ATPS Signer when determining message disposition.

This assertion is based on the fact that the ADMD explicitly endorsed the ATPS Signer. Therefore, a module assessing reputation that is based on DKIM signature verification SHOULD apply the reputation of the Author's ADMD domain instead of, or in addition to, that of the ATPS Signer domain.

6. Relationship to ADSP

[ADSP] defined a protocol by which the owner of an Author Domain can advertise a request to message receivers that messages bearing no valid author signature be treated with suspicion or even discarded.

A Verifier implementing both Author Domain Signing Practices (ADSP) and ATPS MUST test ATPS first. If ATPS indicates a valid delegation, the Verifier MUST act, with respect to ADSP, as though the message has a valid Author Domain Signature (because that's what the delegation means), and no ADSP test is required.

7. Experiment Process

The working group that developed DKIM considered a third-party mechanism such as this one to be controversial, in terms of need and practicality, and decided that an alternative mechanism was sufficient. However, this was not based on actual experience, as there is no specific history on this question. Thus, this experiment was devised.

The experimental protocol described here has been implemented as an extension to DKIM in two software products, one of which is open source and seeing increasingly wide use. It is included there to allow customers of those systems to make use of it if they believe such third-party assertions are useful to the overall DKIM mechanism. Further adoption as part of the experiment is welcome and encouraged.

Use of the protocol and anecdotes of how it affects the overall DKIM experience will be collected by those implementers and the author of this memo. Those participating in the experiment are also advised to observe and report the impact of what is discussed in Section 9.4, especially with respect to MTA latency that may be introduced.

If the response is substantial and positive, advancement along the Standards Track might be warranted.

8. IANA Considerations

This section enumerates requested IANA actions.

8.1. ATPS Tag Registry

IANA has created an Authorized Third-Party Signature (ATPS) Tag Registry, under the DomainKeys Identified Mail (DKIM) Parameters group, to enumerate the tags that are valid for use in ATPS records.

New registrations or updates MUST be made in accordance with the "Specification Required" guidelines described in [IANA]. Such registry changes MUST contain the following information:

1. Name of the tag being registered or updated

2. The document where the specification is created or updated

3. The status of the tag, one of "active" (tag is in current use), "deprecated" (tag is in current use but its use is discouraged), or "historic" (tag is no longer in use)

The registry's initial entries are below:

Tag	Reference	Status
d	[RFC6541]	active
v	[RFC6541]	active

8.2. Email Authentication Methods Registry Update

The following has been added to the Email Authentication Methods registry (in the Email Authentication Parameters group) established by [AUTHRES] as per [IANA]:

Method: dkim-atps

Defined In: [RFC6541]

ptype: header

property: from

value: contents of the [MAIL] From: header field, with comments removed

8.3. Email Authentication Result Names Registry Update

The following have been added to the Email Authentication Result Names registry (in the Email Authentication Parameters group) established by [AUTHRES] as per [IANA]:

Code: none

Existing/New Code: existing

Defined In: [AUTHRES]

Auth Method: dkim-atps

Meaning: No valid DKIM signatures were found on the message bearing "atps" tags.

Code: pass

Existing/New Code: existing

Defined In: [AUTHRES]

Auth Method: dkim-atps

Meaning: An ATPS evaluation was performed, and a valid signature from an authorized third party was found on the message.

Code: fail

Existing/New Code: existing

Defined In: [AUTHRES]

Auth Method: dkim-atps

Meaning: All valid DKIM signatures bearing an "atps" tag either did not reference a domain name found in the RFC5322.From field, or the ATPS test(s) performed failed to confirm a third-party authorization.

Code: temperror

Existing/New Code: existing

Defined In: [AUTHRES]

Auth Method: dkim-atps

Meaning: An ATPS evaluation could not be completed due to some error that is likely transient in nature, such as a temporary DNS error. A later attempt might produce a final result.

Code: permerror

Existing/New Code: existing

Defined In: [AUTHRES]

Auth Method: dkim-atps

Meaning: An ATPS evaluation could not be completed due to some error that is not likely transient in nature, such as a permanent DNS error. A later attempt is unlikely to produce a final result.

8.4. DKIM Signature Tag Specifications Registry

The following have been added to the DKIM Signature Tag Specifications registry (in the DomainKeys Identified Mail (DKIM) Parameters group) established by [DKIM] as per [IANA]:

Type	Reference	Status
atps	[RFC6541]	active
atpsh	[RFC6541]	active

9. Security Considerations

This section discusses potential security issues related to this experimental protocol.

9.1. Hash Selection

The selection of the hash algorithm to be used (see Section 4.1) has security implications, as weaker algorithms have more risk of collision, meaning a second DNS domain name could in theory be constructed to appear to have been authorized by the Author ADMD.

At the time of publication of [DKIM], use of SHA256 was preferred over SHA1 for this reason, though support for both has been maintained. See Section 3.3 of [DKIM] for additional discussion.

9.2. False Privacy

The fact that the authorized third-party domain name is hashed and then encoded with base32 might give some the false sense that the relationship between the two parties is somehow protected. This is not the case. Indeed, the very purpose of this protocol is to make it possible for such relationships to be discovered, so such an obscuration would make that process more difficult without a shared secret delivered out-of-band to message verifiers (which also adds further complexity). Rather, the hash and encode steps are done merely to convert any third-party domain name to a fixed width in the construction of the DNS query.

9.3. Transient Security Failures

Approving a third-party Signer exposes the ADMD to the risk that the third-party Signer becomes compromised and then begins to sign malicious or nuisance messages on behalf of the ADMD. This can obviously reflect negatively on the ADMD, and the impact of this can become more severe as automated domain reputation systems are developed and deployed. Thorough vetting and monitoring practices by ADMDs of third-party Signers will likely need to become the norm.

9.4. Load on the DNS

A Verifier participating in DKIM, ADSP, and ATPS will now issue a number of TXT queries to the DNS equal to as many as one (for the ADSP query) plus the number of signatures on the message (one for each key that is to be verified) plus the number of signatures that validated and that also bear an "atps" tag. This is in addition to any PTR and A queries the MTA might issue at the time the actual message relaying or delivery is initiated. Obviously, this can be burdensome on the DNS at both ends, and waiting for that number of queries to return when they are issued in parallel could trigger timeouts in the MTA.

An alternative that has not yet been explored is the storage of the ATPS data at a specific URL tied to the Author's domain name. This would alleviate pressure on the DNS at the expense of requiring the ADMD to operate a web server (which has its own security implications) and the addition of the establishment of a TCP connection. Moreover, the Verifier would be well advised to implement caching of this data to prevent ATPS from being used as a denial-of-service vector.

See Section 8.5 of [DKIM] for further discussion of DNS-related issues.

10. References

10.1. Normative References

[ABNF] Crocker, D., Ed., and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[AUTHRES] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 5451, April 2009.

[BASE32] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.

[DKIM] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", RFC 6376, September 2011.

[DNS] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[EMAIL-ARCH] Crocker, D., "Internet Mail Architecture", RFC 5598, July 2009.

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[MAIL] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.

10.2. Informative References

[ADSP] Allman, E., Fenton, J., Delany, M., and J. Levine, "DomainKeys Identified Mail (DKIM) Author Domain Signing Practices (ADSP)", RFC 5617, August 2009.

[IANA] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[IDNA] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.

Appendix A. Example Query and Reply

This section presents an example of the use of this protocol to query for a third-party authorization and discusses the interpretation of the result.

Presume a message for which the RFC5322.From domain is "example.com", and it bears two valid signatures, from "one.example.net" and from "two.example.net", each with an "atps" tag whose value is "example.com", and an "atpsh" tag whose value is "sha1". The following actions are taken:

1. A SHA1 hash of "one.example.net" is computed and then converted to printable form using base32 encoding, resulting in the string "QSP4I4D24CRHOPDZ3O3ZIU2KSGS3X6Z6".

2. A TXT query is issued to "QSP4I4D24CRHOPDZ3O3ZIU2KSGS3X6Z6._atps.example.com".

3. If a valid reply arrives, the algorithm stops with [AUTHRES] result "pass". If a DNS error code other than NXDOMAIN is returned, the algorithm stops with a result of "temperror" or "permerror" as appropriate.

4. A SHA1 hash of "two.example.net" is computed and then converted to printable form using base32 encoding, resulting in the string "ZTZGRRV3F45A4U6HLDKBF3ZCOW4V2AJX".

5. A TXT query is issued to "ZTZGRRV3F45A4U6HLDKBF3ZCOW4V2AJX._atps.example.com".

6. If a valid reply arrives, the algorithm stops with [AUTHRES] result "pass". If a DNS error code other than NXDOMAIN is returned, the algorithm stops with a result of "temperror" or "permerror" as appropriate.

7. As there are no valid signatures left to test, the algorithm stops with an "unknown" result.

Appendix B. Choice of DNS RR Type

It was proposed that this work appear within the DNS under a new Resource Record (RR) Type. Although this is possibly an appropriate thing to do, consideration was also given to the fact that major portions of DKIM already use an ASCII-based "tag=value" syntax, and store key and ADSP data in the DNS using TXT resource records. To enable re-use of existing DKIM code, it was decided to re-use the TXT message scheme.

Appendix C. Acknowledgements

The author wishes to acknowledge Dave Crocker, Frank Ellermann, Mark Martinec, and Phil Pennock for their review and constructive criticism of this proposal.

The author also wishes to acknowledge Doug Otis and Daniel Black for their original document, upon which this work was based.

Author's Address

Murray S. Kucherawy
Cloudmark, Inc.
128 King St., 2nd Floor
San Francisco, CA 94107
US

Phone: +1 415 946 3800
EMail: msk@cloudmark.com