pre-workgroup
Internet-Draft
Expires: June 24, 2006

D. Otis
Trend Micro, NSSG

December 21, 2005

Extended Options for DomainKeys Identified Mail (DKIM)
draft-otis-dkim-options-00

Status of this Memo

Copyright Notice

Abstract

This document describes options that extend protections offered by DKIM. These options include Binding-Advice & Role-Assertion, Opaque-Identifier, and an In-Channel check. The Binding-Advice & Role-Assertion offers guidance in isolating the source of a message, in addition to establishing message signature expectations. The Opaque-Identifier (O-ID) offers protection from replay abuse and intra-domain spoofing, even when email-addresses are not associated with the signing-domain. The In-Channel check provides a means to mitigate DNS lookups for avoiding possible message replay abuse.

Table of Contents

1. Introduction

Message signing, as exemplified by DomainKeys Identified Mail (DKIM) [I-D.allman-dkim-base], is a mechanism to allow an assertion of an accountable domain for an email message in transit. The assertion is made by means of a digital signature included within a header, which also validates the integrity of selected headers and message body content subsequent to the signing of the message.

Combining DKIM with an authorization mechanism, referenced from an email-address contained within the message, may result in unintended consequences. The email-address domain owner may be unfairly held accountable for abuse found subsequent to their authorization. As the email-address domain owner often has no administrative oversight or ability to rectify abuse issues, such accountability may place them in peril of having their email refused.

Even when authorization is restricted to a single provider, the email-address domain owner would still be relying upon this provider's diligence, and may be unable to ascertain the cause for refusals or remedy their situation. Such restriction on allowable sources for email also interferes with existing email practices, such as the use of list-servers or sending email using the email-address of one's alma mater. To ensure fair treatment for email-address domain owners, and to minimize the impact upon email practices, the ability to refute messages should not be contingent upon the use of an authorization scheme.

Indicating the results of an authorization that compares an email-address domain to a signing-domain would be unsafe. Domain matching only indicates the email-address is within the same Administrative Unit as the signing-domain. Ambiguity in the display of the email-address and one's limited ability to detect variations from prior messages means such indications may mislead the recipient into erroneously trusting the source of the message.

In addition, the entity directly involved with sending email should be held accountable for abuse. Such an assignment of accountability permits effective and timely remedies, and ensures innocent parties are not inadvertently harmed. For email, such an entity could be discerned by the remote IP address, a verified host name, or the domain used to verify the DKIM-Signature. The DKIM-Signature should be considered an aspect of the message transport, and not necessarily directly associated with message content or any contained email-address.

Restoring trust and establishing the expectation of a signature being present within email messages can be accomplished by way of a

recognition strategy, instead of using an email-address authorization mechanism. Perhaps one of the greatest assets of DKIM would be the enhanced ability to recognize previous email sources. Simple email-address and signing-domain comparisons permit all too common social engineering techniques that are often involved in the spoofing of email-addresses. A recognition strategy can safely highlight those messages emanating from a source specifically recognized by the recipient through a prior message.

The ability to recognize a unique email source is enhanced with the use of the Binding-Advice & Role-Assertion, and the O-ID. The O-ID and In-Channel check can further enhance protections that curtail abusive message replays. The In-Channel check allows a reduction in the overhead associated with abusive message replay protections.

Binding identifiers from a prior correspondent at the behest of the recipient allows indications of recognition without the use of complex and problematic email-address domain authorizations, which may create significant support issues. To support the recognition of a prior correspondent, the MUA could simply highlight those messages from prior correspondents. This approach would offer a higher level of assurance and trust without using any DNS lookups. Following the verification of the DKIM-Signature, the identification of the message source would be contained completely within the message itself.

When assuming legacy MUAs, scant protections are possible by the MTA even using many DNS lookups and registering thousands of look-alike domains. Due to limitations of ensuring the visibility of checked domains, the MTA approach provides an alarmingly low level of email-address protections. There is also a potential for an undesired exposure of email-addresses in the 'i=' parameter.

The O-ID approach could be used to detect when a previous correspondent appears to be from a different source. Without the O-ID, detecting intra-domain spoofing would depend upon the signing-domain verifying the validity of the email-address. The signed message may even advise what other information should be compared against the email-address. While in most cases the collected relationships (bindings) would be made at the behest of the recipient and require their approval, some relationships could be established automatically.

When the email-address domain is within the signing-domain, and when the message advises that these messages should always be signed, then it should be safe to capture this assertion automatically. When signature assurances are captured (cached), the MTA or MUA would be able to detect when a message violated these expected relationships. Before rejecting a message for not having the proper signature, a

check may be made to verify that the signature assurance remains valid.

To recognize the source of a message when there are no assurances being made regarding the email-address, an O-ID that tracks accounts could be added by the signing-domain. This O-ID would become a part of the captured relationship once approved by the recipient. A provision has been added to indicate when the signing role has been delegated. The message from a delegated signer is not allowed to make "broad" recommendations with respect to the scope of a binding. This delegated role will also require the O-ID to equal the left-most label of the DKIM-Signature selector.

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].

Terminology: Terminology conforms to [I-D.crocker-email-arch].

3. Binding-Advice & Role-Assertion

The displayed character-repertoire may be defined by the sender as result of [RFC2047] or [RFC3492]. Even displaying raw puny-code would represent a difficult basis for recognition, especially for recipients who's native language is not based upon ASCII characters. In addition, a large percentage of recipients only see the "display-name" as defined by [RFC2822] (also called the "pretty-name") where the email-address is not normally seen by the recipient.

A safe indication shown to the recipient would be that a message source has been recognized as belonging to a prior correspondent. To help achieve this goal, the signer of the message assists by indicating which aspects of the message's information may be used to isolate the message's source.

Three roles are defined in the following tables. For example, although an MSA is indicated as providing the signature, this role could be delegated to an MUA or another less trusted MSA. Detecting the delegation of a role involves examining the DKIM-Key optional parameters. Whenever the 'g=<email-address>' has an email-address assigned, or the 'w=<sa-validation>' first letter is 'D' then the role should be considered delegated. In the case of a delegated role, the O-ID is derived from the DKIM-Signature 's=<selector>' parameter. When the role is delegated and the 'u=<Opaque-Identifier>' parameter is present, it MUST match that of the left-

most selector label. A "broad" assertion by a Delegated signer is not valid.

Code	Scope of Binding, and Role
w=b	Always signed by MSA, broad ass. across email-domain
w=n	Always signed by MSA, narrow ass. with email-address
w=d	Signed by MSA, broad association across email-domain
w=a	Signed by MSA, narrow association with email-address
w=o	Signed by MSA, association with Opaque-Identifier
none	Signed by MSA, no association assured
w=B	Always signed by Mediator, broad ass. across email-domain
w=N	Always signed by Mediator, narrow ass. with email-address
w=D	Signed by Mediator, broad association across email-domain
w=A	Signed by Mediator, narrow association with email-address
w=O	Signed by Mediator, association with Opaque-Identifier
w=M	Signed by Mediator, no association assured
w=X	Signed by MDA, no association assured

When the DKIM-Signature header field has the option 'w=' with a value of 'b','B','d', or 'D', then the email-address domain associated with the signing-role together with the signing-domain can be used to recognize the source of a message. With a value of 'n','N','a', or 'A', then the entire email-address associated with the signing-role together with the signing-domain should be used to recognize the source of a message. With a value of 'o', or 'O', the O-ID together with the signing-domain can be used to recognize the source of a message. With a value of 'M', 'X' or no 'w=' option (default), just the signing-domain can be used to recognize the source of a message.

When the DKIM-Signature header field has the option 'w=' with a value of 'X', this is used to verify that the message has been accepted by the MDA of the signing-domain and the 'u=' parameter, if present, represents an assessment made by the MDA. To ensure signatures are not misused to perpetrate abusive message replays, the MDA may overlay the 'b=<signature>' of other roles with "!MDA-verified" or "!MDA-invalid". DKIM-Signature header fields containing the 'w=X' option will include other DKIM-Signature header fields containing an "!MDA-verified" signature overlay, in sequential order from the beginning of the message. These additional DKIM-Signature header fields are processed immediately following the processing of the DKIM-Signature header field with the 'w=X' option, and before the remainder of the message. A message with a DKIM-Signature header field signed by a domain of a different Administrative Unit with the 'w=X' option is invalid and SHOULD be rejected.

When the DKIM-Signature header field has the option 'w=' with a value of 'B','N','D', or 'A', then the email-address associated with the DKIM-Signature should be found within a "Resent-*" header field. Each DKIM-Signature should be uniquely associated with a MSA, Mediator, or MDA role. The DKIM-Signature header field added by the MDA or Mediator MUST be removed by the MSA prior to processing the message. When a signature added by an MSA is known by the Mediator to be currently invalid, the DKIM-Signature header field SHOULD be removed or the Mediator may otherwise overlay the 'b=<signature>' with "!Mediator-verified" or "!Mediator-invalid".

Code	Binding	Assurances	Validation	Role
w=b	email-domain	always signed	DKIM key	MSA
w=n	email-address	always signed	DKIM key	MSA
w=d	email-domain	none	none	MSA
w=a	email-address	none	none	MSA
w=o	Opaque-Identifier	none	none	MSA
none	none	none	none	MSA
w=B	email-domain	always signed	DKIM key	Mediator
w=N	email-address	always signed	DKIM key	Mediator
w=D	email-domain	none	none	Mediator
w=A	email-address	none	none	Mediator
w=O	Opaque-Identifier	none	none	Mediator
w=M	none	none	none	Mediator
w=X	none	none	none	MDA

When the DKIM-Signature header field has the option 'w=' with a value of 'n', or 'b', and the email-address domain is within the signing-domain as denoted by the 'd=<domain>' parameter, then the assurance of a signature for this domain can be automatically cached. The cached information should include both the domain where an assurance has been made, and the label for a record used to confirm the continued status of the assurance. A parameter within the DKIM-Key can be used to consolidate where assurances are confirmed when multiple DKIM-Keys are being used. When there are no parameters added to the DKIM-Key, the default signature-assurance validation location would be determined by the 'd=<domain>' and parameters 's=<selector>'. The left-most label within the selector would be used as follows:

"<lm-selector>._dkim-sa.<domain>".

When the 'w=' option is present within the DKIM-Key, the value of this parameter modifies the signature-assurance validation location

to be:

"<sa-validation>._dkim-sa.<domain>".

The 'w=<sa-validation>' option would be composed of 1 to 63 characters within the DKIM-Key and used to consolidate signature assurances. The operation of this signature-assurance validation record mechanism would take the form of a single A record lookup where the existence of the record would validate a cached assurance.

```
<sa-validation> ::= <role> [ [ <ldh-str> ] <let-dig> ]
<ldh-str> ::= <let-dig-hyp> | <let-dig-hyp> <ldh-str>
<let-dig-hyp> ::= <let-dig> | "-"
<let-dig> ::= <letter> | <digit>
<letter> ::= %x41-5A / %x61-7A
<role> ::= "T" | "D"
  ; Trusted or Delegated role
<digit> ::= %x30-39
<sa-validation>._dkim-sa.<signing-domain> IN A 127.0.0.2
```

4. Opaque-Identifier

The Opaque-Identifier (O-ID) is an option that supports two different mechanisms. One mechanism isolates the source of a message to a specific account as denoted by the Binding-Advice & Role-Assertion. The other mechanism provides a means to revoke messages being abusively replayed. An O-ID added to the signature header MUST also be a valid domain name label. The term 'opaque' means only the domain creating the identifier understands the associations indicated in Binding-Advice & Role-Assertion. There are two modes for creating the O-ID. One mode would make the O-ID persistent with the account used to access the signing-domain, and the other could be sequential for cases where an account is not involved. A prefix added to a sequential O-ID prevents collisions with identifiers used for accounts.

If an identifier were added to an unsigned message, this would invite forgery and therefore offer little value. A standardized O-ID, included within the validated content of a signed message, would offer significant value. A persistent O-ID would be most useful and could be derived from the access server that authenticates the account being used.

A sequential O-ID may be appropriate when distributing bulk mailings. To identify abusers that may attempt to stage replay attacks, having a unique identifier for each recipient could prove helpful. These

replay attacks could be done using the unchanged content of the message, but sent to recipients that would consider the information to be unsolicited. The reason for such a replay attack may be to damage the reputation of the signing-domain.

The persistent O-ID would greatly aid the correlation of abuse and the locating of compromised systems. This identifier could be effective against systems compromised by Trojan programs, stolen passwords, and cracked wireless access points, among many other nefarious methods. Abuse reports that catalog signed messages and that are correlated with a persistent O-ID would provide incontrovertible evidence of where the source of a problem exists. The publishing of the revocation record for the O-ID would also provide feedback that actions were taken to rectify a policy breach.

In odd cases where an In-Channel check fails, a single lookup of a revocation record for the O-ID returning no record would be an indication that this particular O-ID is still authorized by the signing-domain. This mechanism would be most valuable in those cases where the message may have been forwarded, such as at the typical alma mater, or where a mailing list opts to also forward signed messages unaltered.

If there is a problem, the signature would offer the name of the most capable domain able to remedy abuse. People can still safely use their forwarding email accounts given to them by their school or society. Mailing lists would be given a strong identifier upon which to grant the replication of messages. Complaints would also likely be directed to those most able to curtail future episodes of bad behavior, i.e. the provider of the abusive account!

Within the signature header, a 'u=<Opaque-Identifier>' parameter or within the DKIM-Key, a 'w=<sa-assurance>' parameter where the first letter is 'D' a would indicate the use of an O-ID. The operation of this revocation record mechanism takes the form of a single A record lookup where the return of a record indicates the O-ID has been revoked. The O-ID would be composed of 1 to 63 characters and select a record in this fashion:

```
<Opaque-Identifier> ::= <mode> [ [ <ldh-str> ] <let-dig> ]
<ldh-str> ::= <let-dig-hyp> | <let-dig-hyp> <ldh-str>
<let-dig-hyp> ::= <let-dig> | "-"
<let-dig> ::= <letter> | <digit>
<letter> ::= %x41-5A / %x61-7A
<mode> ::= "P" |  "S"
  ; Persistent and Sequential O-ID assignment
<digit> ::= %x30-39
<Opaque-Identifier>._dkim-or.<signing-domain> IN A 127.0.0.2
```

When the first letter in the O-ID is 'P,' this represents an identifier where the portion of the identifier to the right of the leftmost '-' character is persistent with the account used to obtain access. When the first letter is 'S,' then no portion of the identifier can be used to isolate which account was used to obtain access.

When the signing-domain has not revoked authorization for the O-ID, no record would be returned and the remote DNS cache would retain the absence of this record for a brief period of time, see [RFC2308]. For the majority of cases, where messages are obtained directly from the signing-domain, confirmation of an In-Channel check allows the O-ID revocation check to be skipped.

The O-ID revocation check would be performing nearly an identical lookup now ubiquitously done to investigate the status of the SMTP client's IP address against a DNS black-hole list. Those addresses or identifiers that warrant refusal are granted a long lived address record to ensure their immediate refusal and limit DNS traffic resulting from abusive sources. Otherwise, not offering a record allows for the prompt cessation of an O-ID's authorization when the situation regarding a particular identifier changes. The Time-To-Live for negative DNS caching may be determined by the recipient, or represent the lesser of the SOA TTL or the SOA MINIMUM field, depending upon the recipient's implementation.

5. In-Channel Check

There are two methods that can be used to ascertain whether a message is In-Channel. In-Channel would be when the RCPT TO list has been specified by or sourced by the originating Administrative Unit. One method uses a hash of the initial [RFC2821] RCPT TO: email-address list. The other method verifies the EHLO using a DNS lookup for an address record or CSV-CSA record as defined in [I-D.crocker-csv-csa]. When the signing-domain as noted in the DKIM-Signature 'd=<domain>' parameter are within the verified EHLO domain name, the message could be said to be In-Channel. Another method may use a [RFC2821] RCPT TO: email-address hash parameter stored within the DKIM-Signature to confirm that the RCPT TO list has not been altered.

When the message is determined to be In-Channel, and an O-ID option is being used, checking for O-ID revocation may be skipped. When O-ID revocation should be checked, the receiving SMTP server may issue an SMTP 450 temporary error and delay acceptance for a few minutes. Once the receiving SMTP server decides enough time has elapsed from the initial delivery attempt for the specific message, a O-ID revocation check would be made instead. If the O-ID authorization has not been revoked, the message may be accepted.

When the 'm=' parameter is included, an SHA-1 hash algorithm defined in [RFC3174] is used to hash all [RFC2821] RCPT TO: email-addresses in sequence from left to right and first to last. The hash will include only the [RFC2821] RCPT TO: email-addresses, and to obfuscate the use of a BCC header, the hash may be initialized by a special SMTP extension MF-SALT. The result of the hash is stored in Base 64 within the DKIM-Signature 'm=<mailfrom-hash>' parameter. When the MF-SALT extension has been allowed, a RCPT TO parameter may return an SMTP extension MF-SALT-?????????????? where the fourteen '?' are replaced by "URL and Filename safe" Base 64 Alphabet characters as defined in [RFC3548] representing an 84 bit random number. When the MF-SALT parameter is found within the initial RCPT TO parameter, without a binary conversion, the fourteen Base 64 Alphabet characters are hashed first, followed by the RCPT TO: email-addresses. When the MF-SALT parameter is not present, just the RCPT TO: list may have been hashed.

6. IANA Considerations

The SMTP extension MF-SALT will require registration by IANA.

Use of the _dkim-sa and _dkim-or prefix in DNS records will require registration by IANA.

To avoid conflicts, tag names for the DKIM-Signature header and key records the following should be added to those registered with IANA.

Tag values for the "w=", "u=", and "m=" tags in the DKIM-Signature header, and the "w=", tags in key records should be registered with IANA for the same reason.

7. Security Considerations

This document describes options that can be used with DomainKeys Identified Mail (DKIM) to improve upon the secure use of this mechanism.

8. References

8.1 Normative References

[I-D.crocker-csv-csa]
Crocker, D., "Client SMTP Authorization (CSA)", draft-crocker-csv-csa-00 (work in progress), October 2005.

[I-D.crocker-email-arch]
Crocker, D., "Internet Mail Architecture", draft-crocker-email-arch-04 (work in progress),

March 2005.

[RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, November 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2308] Andrews, M., "Negative Caching of DNS Queries (DNS NCACHE)", RFC 2308, March 1998.

[RFC2821] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821, April 2001.

[RFC2822] Resnick, P., "Internet Message Format", RFC 2822, April 2001.

[RFC3174] Eastlake, D. and P. Jones, "US Secure Hash Algorithm 1 (SHA1)", RFC 3174, September 2001.

[RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, March 2003.

[RFC3548] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 3548, July 2003.

8.2 Informative References

[I-D.allman-dkim-base]
Allman, E., "DomainKeys Identified Mail (DKIM)", draft-allman-dkim-base-01 (work in progress), October 2005.

Author's Address

Douglas Otis
Trend Micro, NSSG
1737 North First Street, Suite 680
San Jose, CA 95112
USA

Phone: +1.408.453.6277
Email: doug_otis@trendmicro.com

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.