Network Working Group
Request for Comments: 4406
Category: Experimental

J. Lyon
Microsoft Corp.
M. Wong
pobox.com

April 2006

Sender ID: Authenticating E-Mail

Status of This Memo

This memo defines an Experimental Protocol for the Internet community. It does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited.

Copyright Notice

IESG Note

The following documents (RFC 4405, RFC 4406, RFC 4407, and RFC 4408) are published simultaneously as Experimental RFCs, although there is no general technical consensus and efforts to reconcile the two approaches have failed. As such, these documents have not received full IETF review and are published "AS-IS" to document the different approaches as they were considered in the MARID working group.

The IESG takes no position about which approach is to be preferred and cautions the reader that there are serious open issues for each approach and concerns about using them in tandem. The IESG believes that documenting the different approaches does less harm than not documenting them.

Note that the Sender ID experiment may use DNS records that may have been created for the current SPF experiment or earlier versions in this set of experiments. Depending on the content of the record, this may mean that Sender-ID heuristics would be applied incorrectly to a message. Depending on the actions associated by the recipient with those heuristics, the message may not be delivered or may be discarded on receipt.

Participants relying on Sender ID experiment DNS records are warned that they may lose valid messages in this set of circumstances. Participants publishing SPF experiment DNS records should consider

the advice given in section 3.4 of RFC 4406 and may wish to publish both v=spf1 and spf2.0 records to avoid the conflict.

Participants in the Sender-ID experiment need to be aware that the way Resent-* header fields are used will result in failure to receive legitimate email when interacting with standards-compliant systems (specifically automatic forwarders which comply with the standards by not adding Resent-* headers, and systems which comply with RFC 822 but have not yet implemented RFC 2822 Resent-* semantics). It would be inappropriate to advance Sender-ID on the standards track without resolving this interoperability problem.

The community is invited to observe the success or failure of the two approaches during the two years following publication, in order that a community consensus can be reached in the future.

Abstract

Internet mail suffers from the fact that much unwanted mail is sent using spoofed addresses -- "spoofed" in this case means that the address is used without the permission of the domain owner. This document describes a family of tests by which SMTP servers can determine whether an e-mail address in a received message was used with the permission of the owner of the domain contained in that e-mail address.

Table of Contents

1. Introduction 3
1.1. Conventions Used in This Document 4
2. Problem Statement 4
3. SPF 2.0 Records 5
3.1. Version and Scope 5
3.1.1. Minor Version 6
3.2. Multiple Records 6
3.3. Positional Modifiers 7
3.4. Compatibility 8
4. Decision Model 8
4.1. Arguments 9
4.2. Results 9
4.3. Record Lookup 9
4.4. Record Selection 9
5. Actions Based on the Decision 10
5.1. Neutral, None, SoftFail, or PermError 11
5.2. Pass 11
5.3. Fail 11
5.4. TempError 11
6. Security Considerations 11
6.1. DNS Attacks 12
6.2. TCP Attacks 12
6.3. Forged Sender Attacks 12
6.4. Address Space Hijacking 12
6.5. Malicious DNS Attacks on Third Parties 13
7. Implementation Guidance 13
7.1. Simple E-Mailers 14
7.2. E-Mail Forwarders 14
7.3. Mailing List Servers 15
7.4. Third-Party Mailers 15
7.5. MUA Implementers 15
8. Acknowledgements 16
9. References 17
9.1. Normative References 17
9.2. Informative References 17

1. Introduction

Today, a huge majority of unwanted e-mail contains headers that lie about the origin of the mail. This is true of most spam and substantially all of the virus e-mail that is sent.

This document describes a mechanism such that receiving Mail Transfer Agents (MTAs), Mail Delivery Agents (MDAs), and/or Mail User Agents (MUAs) can recognize mail in the above category and take appropriate action. For example, an MTA might refuse to accept a message, an MDA

might discard a message rather than placing it into a mailbox, and an MUA might render that message in some distinctive fashion.

In order to avoid further fragmentation of the Internet e-mail system, it is desirable that the Internet community as a whole come to a consensus as to what mail senders should do to make their mail appear non-spoofed, and how mail receivers should determine whether mail is spoofed. On the other hand, it is not necessary to reach a consensus regarding the actions that various parties take once a message has been determined to be spoofed. This can be done unilaterally -- one agent might decide to discard a spoofed message whereas another decides to add a disclaimer.

This document defines a pair of closely-related tests. One validates a message's Purported Responsible Address (PRA) as defined in [RFC4407]. The other validates a message's Reverse-Path (also known as MAIL-FROM address) as defined in [RFC4408].

An e-mail sender complying with this specification SHOULD publish information for both tests, and SHOULD arrange that any mail that is sent will pass both tests. An e-mail receiver complying with this specification SHOULD perform at least one of these tests.

1.1. Conventions Used in This Document

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119].

2. Problem Statement

Briefly stated, the mechanisms of this document allow one to answer the following question:

> When a message is transferred via SMTP between two unrelated parties, does the SMTP client host have permission to send mail on behalf of a mailbox referenced by the message?

As seen from the question, this mechanism applies to unrelated parties: It is useful at the point where a message passes across the Internet from one organization to another. It is beyond the scope of this document to describe authentication mechanisms that can be deployed within an organization.

The PRA version of the test seeks to authenticate the mailbox associated with the most recent introduction of a message into the mail delivery system. In simple cases, this is who the mail is from. However, in the case of a third-party mailer, a forwarder, or a

mailing list server, the address being authenticated is that of the third party, the forwarder, or the mailing list.

On the other hand, the MAIL-FROM version of the test seeks to authenticate the mailbox that would receive Delivery Status Notifications (DSNs, or bounces) for the message. In simple cases, this too is who the mail is from. However, third-party mailers, forwarders, and mailing list servers MUST specify an address under their control, and SHOULD arrange that DSNs received at this address are forwarded to the original bounce address.

In both cases, the domain associated with an e-mail address is what is authenticated; no attempt is made to authenticate the local-part. A domain owner gets to determine which SMTP clients speak on behalf of addresses within the domain; a responsible domain owner should not authorize SMTP clients that will lie about local parts.

In the long run, once the domain of the sender is authenticated, it will be possible to use that domain as part of a mechanism to determine the likelihood that a given message is spam, using, for example, reputation and accreditation services. (These services are not the subject of the present mechanism, but it should enable them.)

3. SPF 2.0 Records

Domains declare which hosts are and are not authorized to transmit e-mail messages on their behalf by publishing Sender Policy Framework (SPF) records in the Domain Name System. [RFC4408] defines a format for these records identified by the version prefix "v=spf1". This section defines an amended format, identified by the version prefix "spf2.0", that allows sending domains to explicitly specify how their records should be interpreted, and provides for additional extensibility. Sending domains MAY publish either or both formats.

Since the two formats are identical in most respects, the following subsections define the "spf2.0" format relative to [RFC4408].

3.1. Version and Scope

Under Sender ID, receiving domains may perform a check of either the PRA identity or the MAIL-FROM identity. Sending domains therefore require a method for declaring whether their published list of authorized outbound e-mail servers can be used for the PRA check, the MAIL-FROM check, or both.

This section replaces the definition of the version identifier in Section 4.5 of [RFC4408] and adds the concept of SPF record scopes.

SPF records begin with a version identifier and may also include a scope:

```
record      = version terms *SP
version     = "v=spf1" | ( "spf2." ver-minor scope)
ver-minor   = 1*DIGIT
scope       = "/" scope-id *( "," scope-id )
scope-id    = "mfrom" / "pra" / name
```

For example, the SPF record

```
spf2.0/mfrom,pra +mx +ip4:192.168.0.100 -all
```

defines an SPF record that can be used for either MAIL FROM or PRA checks.

This document only defines the existence of two scopes: "mfrom" and "pra". The details of these two scopes are defined in other documents: "mfrom" is defined in [RFC4408]; "pra" is defined in [RFC4407].

Other scopes may be defined by future documents only. There is no registry for scopes. A scope definition must define what it identifies as the sending mailbox for a message, how to extract that information from a message, how to determine the initial arguments for the check_host() function, and what the compliant responses to the result are. This ensures that domains with published records and mail receiver agree on the semantics of the scope.

A compliant domain SHOULD publish authorizations for every defined scope.

3.1.1. Minor Version

All published records that use the "spf2" version identifier MUST start with "spf2.0". This document only specifies records with a minor version of "0".

Future versions of this document may define other minor versions to be used.

3.2. Multiple Records

A domain MAY publish multiple SPF 2.0 records, provided that each scope appears in at most one SPF 2.0 record. In addition, a domain MAY also publish an SPF record that uses the "v=spf1" version identifier defined in [RFC4408]. The selection rules in Section 4.4 define the precedence of these records.

3.3. Positional Modifiers

This section replaces Section 4.6.3 of [RFC4408] and adds the concept of positional modifiers.

Modifiers are key/value pairs that affect the evaluation of the check_host() function.

Modifiers are either global or positional:

- Global modifiers MAY appear anywhere in the record, but SHOULD appear at the end, after all mechanisms and positional modifiers.

- Positional modifiers apply only to the mechanism they follow. It is a syntax error for a positional modifier to appear before the first mechanism.

Modifiers of either type are also either singular or multiple:

- Singular modifiers may appear only once in the record if they are global, or once after each mechanism if they are positional.

- Multiple modifiers may appear multiple times in the record if they are global, or multiple times after each mechanism if they are positional.

A modifier is not allowed to be defined as both global and positional.

The modifiers "redirect" and "exp" described in Section 6 of [RFC4408] are global and singular.

Ordering of modifiers does not matter, except that:

1. positional modifiers must appear after the mechanism they affect and before any subsequent mechanisms; and

2. when a multiple modifier appears more than one time, the ordering of the appearances may be significant to the modifier.

Other than these constraints, implementations MUST treat different orders of modifiers the same. An intended side effect of these rules is that modifiers cannot be defined that modify other modifiers.

These rules allow an implementation to correctly pre-parse a record. Furthermore, they are crafted to allow the parsing algorithm to be stable, even when new modifiers are introduced.

Modifiers that are unrecognized MUST be ignored. This allows older implementations to handle records with modifiers that were defined after they were written.

3.4. Compatibility

Domain administrators complying with this specification are required to publish information in DNS regarding their authorized outbound e-mail servers. [RFC4408] describes a format for this information identified by the version prefix "v=spf1". Many domains have published information in DNS using this format. In order to provide compatibility for these domains, Sender ID implementations SHOULD interpret the version prefix "v=spf1" as equivalent to "spf2.0/mfrom,pra", provided no record starting with "spf2.0" exists.

Administrators who have already published "v=spf1" records SHOULD review these records to determine whether they are also valid for use with PRA checks. If the information in a "v=spf1" record is not correct for a PRA check, administrators SHOULD publish either an "spf2.0/pra" record with correct information or an "spf2.0/pra ?all" record indicating that the result of a PRA check is explicitly inconclusive.

4. Decision Model

Sender ID enables receiving e-mail systems to answer the following question:

Given an e-mail message, and given an IP address from which it has been (or will be) received, is the SMTP client at that IP address authorized to send that e-mail message?

This question will usually be asked by an SMTP server as part of deciding whether to accept an incoming mail message. However, this question could also be asked later by a different party. An MUA, for example, could use the result of this question to determine how to file or present a message.

There are three steps to answering this question:

1. From an e-mail message, extract the address to verify. The PRA variant of this test does so as specified in [RFC4407], or alternatively, using the submitter address as specified in [RFC4405]. The MAIL FROM variant of this test does so as specified in [RFC4408].

2. Extract the domain part of the address determined in step 1.

3. Call the check_host() function defined in [RFC4408] and modified by the following subsections.

If the Sender ID check is being performed by an MTA as part of receiving an e-mail message, and it cannot determine an address in step 1 above (because the message or address is malformed), then the message SHOULD be rejected with error "550 5.7.1 Missing Purported Responsible Address" or error "550 5.7.1 Missing Reverse-Path address".

4.1. Arguments

Sender ID modifies the check_host() function by the addition of a scope parameter. Thus, for Sender ID the check_host() function is called passing the following parameters:

a. A scope of "pra" (for the PRA variant of the test), or "mfrom" (for the MAIL FROM variant of the test).
b. The IP address (either IPv4 or IPv6) from which the message is being or has been received.
c. The domain from step 2 above.
d. The address from step 1 above.

4.2. Results

The result of the check_host() function is one of the values "Neutral", "Pass", "Fail", "SoftFail", "None", "TempError", or "PermError". Section 5 describes how these results are used by MTAs receiving messages. This specification imposes no requirements on parties performing this test in other environments.

4.3. Record Lookup

SPF records are looked up in DNS in accordance with Section 4.4 of [RFC4408].

When performing the PRA version of the test, if the DNS query returns "non-existent domain" (RCODE 3), then check_host() exits immediately with the result "Fail".

4.4. Record Selection

This section replaces the record selection steps described in Section 4.5 of [RFC4408].

Starting with the set of records that were returned by the lookup, record selection proceeds in these steps:

1. If any records of type SPF are in the set, then all records of type TXT are discarded.

2. Records that do not begin with proper version and scope sections are discarded. The version section for "spf2" records contains a ver-minor field that is for backward-compatible future extensions. This field must be well formed for a record to be retained, but is otherwise ignored.

3. Records that use the "spf2" version identifier and do not have a scope-id that matches <scope> are discarded. Note that this is a complete string match on the scope-id tokens: If <scope> is "pra", then the record starting "spf2.0/mfrom,prattle,fubar" would be discarded, but a record starting "spf2.0/mfrom,pra,fubar" would be retained.

4. If the lookup returned two records, one containing the "v=spf1" version identifier and the other containing the "spf2" version identifier, the "spf2" version takes precedence for the desired scope-id. If the "spf2" record does not contain the desired scope-id, then the "v=spf1" record is selected.

5. If an "spf2" record does not contain the desired scope-id and there is no "v=spf1" record for the domain, then no record is selected.

After the above steps, there should be one record remaining and evaluation can proceed. If there are two or more records remaining, then check_host() exits immediately with the error "PermError".

If there are no matching records remaining after the initial DNS query or any subsequent optional DNS queries, then check_host() exits immediately with the result "None".

5. Actions Based on the Decision

When the Sender ID test is used by an SMTP server as part of receiving a message, the server should take the actions described by this section.

The check_host() function returns one of the following results. See [RFC4408] for the meaning of these results.

5.1. Neutral, None, SoftFail, or PermError

An SMTP server receiving one of these results SHOULD NOT reject the message for this reason alone, but MAY subject the message to heightened scrutiny by other measures, and MAY reject the message as a result of this heightened scrutiny.

Such additional security measures MAY take into account that a message for which the result is "SoftFail" is less likely to be authentic than a message for which the result is "Neutral".

5.2. Pass

An SMTP server receiving this result SHOULD treat the message as authentic. It may accept or reject the message depending on other policies.

5.3. Fail

When performing the Sender ID test during an SMTP transaction, an MTA that chooses to reject a message receiving this result SHOULD reject the message with a "550 5.7.1 Sender ID (xxx) yyy - zzz" SMTP error, where "xxx" is replaced with "PRA" or "MAIL FROM", "yyy" is replaced with the additional reason returned by the check_host() function, and "zzz" is replaced with the explanation string returned by the check_host() function.

When performing the Sender ID test after accepting an e-mail message for delivery, an MTA that chooses to reject a message receiving this result SHOULD NOT deliver the message. Instead, it should create a DSN message, consistent with the usual rules for DSN messages.

5.4. TempError

An SMTP server receiving this result MAY reject the message with a "450 4.4.3 Sender ID check is temporarily unavailable" error code. Alternatively, an SMTP server receiving this result MAY accept a message and optionally subject it to heightened scrutiny by other anti-spam measures.

6. Security Considerations

This entire document describes a new mechanism for mitigating spoofed e-mail, which is today a pervasive security problem in the Internet.

Assuming that this mechanism is widely deployed, the following sections describe counter attacks that could be used to defeat this mechanism.

6.1. DNS Attacks

The new mechanism is entirely dependent on DNS lookups, and is therefore only as secure as DNS. An attacker bent on spoofing messages could attempt to get his messages accepted by sending forged answers to DNS queries.

An MTA could largely defeat such an attack by using a properly paranoid DNS resolver. DNS Security (DNSSEC) may ultimately provide a way to completely neutralize this class of attacks.

6.2. TCP Attacks

This mechanism is designed to be used in conjunction with SMTP over TCP. A sufficiently resourceful attacker might be able to send TCP packets with forged from-addresses, and thus execute an entire SMTP session that appears to come from somewhere other than its true origin.

Such an attack requires guessing what TCP sequence numbers an SMTP server will use. It also requires transmitting completely in the blind -- the attack will be unable to hear any of the server's side of the conversation.

Attacks of this sort can be ameliorated if IP gateways refuse to forward packets when the source address is clearly bogus.

6.3. Forged Sender Attacks

This mechanism chooses an address to validate either from one of a number of message headers or from the RFC 2821 MAIL command, and then uses that address for validation. A message with a true Resent-From header or Return-Path, but a forged From header, will be accepted. Since many MUAs do not display all of the headers of received messages, the message will appear to be forged when displayed.

In order to neutralize this attack, MUAs will need to start displaying at least the address that was verified. In addition, MTAs could subject messages to heightened scrutiny when the validated address differs from the From header.

6.4. Address Space Hijacking

This mechanism assumes the integrity of IP address space for determining whether a given client is authorized to send messages from a given PRA. In addition to the TCP attack given in Section 6.2, a sufficiently resourceful attacker might be able to alter the IP routing structure to permit two-way communication using a

specified IP address. It would then be possible to execute an SMTP session that appears to come from an authorized address, without the need to guess TCP sequence numbers or transmit in the blind.

Such an attack might occur if the attacker obtained access to a router that participates in external BGP routing. Such a router could advertise a more specific route to a rogue SMTP client, temporarily overriding the legitimate owner of the address.

6.5. Malicious DNS Attacks on Third Parties

There is class of attacks in which an attacker A can entice a participant P to send a malicious message to a victim V.

These attacks are undertaken by A citing the address of V in the SMTP MAIL FROM request and then by causing P to generate (or invoke the generation of) a Delivery Status Notification 'bounce' message (RFC3464), which is sent to the victim V.

The attacker relies upon it being common practice to copy the original message into the 'bounce' report, thereby causing the malice to be sent onward to V.

This mode of attack has the advantages (to the attacker) of obfuscating the location of the host from which the attack was mounted, and of possibly damaging the reputation of P by making it appear that P originated or was an active participant in the sending of the malicious message.

In current practice, A causes P to cause the 'bounce' by addressing the original message to a nonexistent recipient.

Sender ID enables a new variant of this attack.

In this variant, the attacker A sends a message whose PRA (Section 4) is selected by the attacker to be such that, when P undertakes the Sender ID test, a Fail will result (Section 5.3).

The message will be rejected (as the attacker intended) and a malicious 'bounce' message may be generated and sent to the victim V.

7. Implementation Guidance

This section describes the actions that certain members of the Internet e-mail ecosystem must take to be compliant with this specification.

7.1. Simple E-Mailers

A domain that injects original e-mail into the Internet, using its own name in From headers, need do nothing to be compliant. However, such domains SHOULD publish records in DNS as defined by [RFC4408] and this specification.

In the majority of cases, the domain's published information will be the same for both the PRA and MAIL FROM variants of this test. In this case, domains SHOULD publish their information using an SPF record with the prefix "v=spf1". Doing so will render their published information usable by the older SPF protocol, too. (See [RFC4408] for information on the SPF protocol.)

7.2. E-Mail Forwarders

In order to pass the PRA variant of the test, a program that forwards received mail to other addresses MUST add an appropriate header that contains an e-mail address that it is authorized to use. Such programs SHOULD use the Resent-From header for this purpose.

In order to pass the MAIL FROM variant of the test, a program that forwards received mail to other addresses MUST alter the MAIL FROM address to an address under its control. Should that address eventually receive a DSN relating to the original message, that DSN SHOULD be forwarded to the original MAIL FROM address. However, if this altered address receives any messages other than DSNs related to the original message, these messages MUST NOT be forwarded to the original MAIL FROM address; they SHOULD be refused during an SMTP transaction.

In addition, e-mail forwarders SHOULD publish Sender ID records for their domains, and SHOULD use MTAs for which the Sender ID check yields a "pass" result.

Some of today's forwarders already add an appropriate header (although many of them use Sender rather than Resent-From.) Most of them do not perform the address-rewriting specified above.

Note that an e-mail forwarder might receive a single message for two or more recipients, each of whom requests forwarding to a new address. In this case, the forwarder's MTA SHOULD transmit the message to each new recipient individually, with each copy of the message containing a different newly inserted Resent-From header field.

7.3. Mailing List Servers

In order to pass the PRA variant of the test, a mailing list server MUST add an appropriate header that contains an e-mail address that it is authorized to use. Such programs SHOULD use the Resent-From header for this purpose.

In order to pass the MAIL FROM variant of the test, a mailing list server MUST alter the MAIL FROM address to an address under its control.

In addition, mailing list servers SHOULD publish Sender ID records for their domains, and SHOULD use MTAs for which the Sender ID check yields a "pass" result.

Most of today's mailing list software already adds an appropriate header (although most of them use Sender rather than Resent-From), and most of them already alter the MAIL FROM address.

7.4. Third-Party Mailers

In order to pass the PRA variant of this test, a program that sends mail on behalf of another user MUST add an appropriate header that contains an e-mail address that it is authorized to use. Such programs SHOULD use the Sender header for this purpose.

In order to pass the MAIL FROM variant of this test, a program that sends mail on behalf of another user MUST use a MAIL FROM address that is under its control. Defining what the program does with any mail received at that address is beyond the scope of this document.

In addition, third-party mailers, servers SHOULD publish Sender ID records for their domains, and SHOULD use MTAs for which the Sender ID check yields a "pass" result.

Many, but not all, of today's third-party mailers are already compliant with the PRA variant of the test. The extent to which mailers are already compliant with the MAIL FROM variant of this test is unknown.

7.5. MUA Implementers

When displaying a received message, an MUA SHOULD display the purported responsible address as defined by this document whenever that address differs from the RFC 2822 From address. This display SHOULD be in addition to the RFC 2822 From address.

When a received message contains multiple headers that might be used for the purported responsible address determination, an MUA should consider displaying all of them. That is, if a message contains several Resent-From's, a Sender, and a From, an MUA should consider displaying all of them.

Sender ID also does not validate the display name that may be transmitted along with an e-mail address. The display name is also vulnerable to spoofing and other forms of attacks. In order to reduce the occurrence and effectiveness of such attacks, MUA implementers should consider methods to safeguard the display name. This could include the following:

* Not presenting the display name to the user at all, or not presenting the display name unless the corresponding e-mail address is listed in the user's address book.

* Treating as suspicious any e-mail where the display name is itself in the form of an e-mail address, especially when it differs from the actual e-mail address in the header.

* Making it clear to users that the e-mail address has been checked rather than the display name.

8. Acknowledgements

This design is based on earlier work published in 2003 in [RMX] and [DMP] drafts (by Hadmut Danisch and Gordon Fecyk, respectively). The idea of using a DNS record to check the legitimacy of an e-mail address traces its ancestry to "Repudiating Mail From" draft by Paul Vixie [Vixie] (based on suggestion by Jim Miller) and to "Domain-Authorized SMTP Mail" draft by David Green [Green], who first introduced this idea on the namedroppers mailing list in 2002.

The current document borrows heavily from each of the above, as well as earlier versions of [RFC4408] and [CallerID], and incorporates ideas proposed by many members of the MARID working group. The contributions of each of the above are gratefully acknowledged.

9. References

9.1. Normative References

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

[RFC4405] Allman E. and H. Katz, "SMTP Service Extension for Indicating the Responsible Submitter of an E-Mail Message", RFC 4405, April 2006.

[RFC4407] Lyon, J., "Purported Responsible Address in E-Mail Messages", RFC 4407, April 2006.

[RFC4408] Wong, M. and W. Schlitt, "Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail", RFC 4408, April 2006.

9.2. Informative References

[CallerID] Microsoft Corporation, Caller ID for E-Mail Technical Specification, http://www.microsoft.com/mscorp/safety/technologies/senderid/resources.mspx

[DMP] Fecyk, G., "Designated Mailers Protocol", http://www.pan-am.ca/dmp/draft-fecyk-dmp-01.txt, December 2003.

[Green] David Green, "Mail-Transmitter RR", http://ops.ietf.org/lists/namedroppers/namedroppers.2002/msg00656.html, June 2002.

[RMX] H. Danisch, "The RMX DNS RR and method for lightweight SMTP sender authorization", http://www.danisch.de/work/security/txt/draft-danisch-dns-rr-smtp-04.txt

[Vixie] Paul Vixie, "Repudiating Mail From", http://ops.ietf.org/lists/namedroppers/namedroppers.2002/msg00658.html, June 2002.

Authors' Addresses

Jim Lyon
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
USA

EMail: jimlyon@microsoft.com

Meng Weng Wong
Singapore

EMail: mengwong@dumbo.pobox.com

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).