Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 28, 2012

H. Santos, Ed.
Santronics Software, Inc.
E. Harris
puremagic.com

April 26, 2012

SMTP Service Extension for Greylisting Operations
draft-santos-smtpgrey-02

Abstract

GREYLIST is a SMTP extension to formalize the widely supported Greylisting mail filtering method and to help support SMTP rejected transports by following a new formal structured 4yz server temporary rejection response by including a "retry=time-delay" tag string which SMTP clients can use to optimize the rescheduling of the mail delivery attempts. With adoption, network overhead reduction in wasteful mail delivery attempts will be realized.

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 October 28, 2012.

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

In 2003, a non-IETF technology called GreyListing was invented by Evan Harris [HARRIS] as a very effective method of enhancing the abilities of SMTP [RFC5321] mail systems to limit the amount of unwanted, abusive mail that they receive and deliver to their users. Mail systems supporting GreyListing has grown over the years to become a "pseudo standard" among many SMTP operations.

This specification provides a formal IETF specification to the Greylisting framework, learned practices and introduces a SMTP extension to reduce the network, traffic overheads and mail delivery delays associated with SMTP Greylisting operations.

1.1. Background

Greylisting was originally tested on a few small scale mail hosts (less than 100 users, though with a fairly diverse set of senders from all over the world, and volumes over 10,000 email attempts a day). Currently, Greylisting is in use on many mail servers, including ones processing several millions of messages per day. It was designed to be scalable and marginal impact to both administrators and users, and should be acceptable for use on a wide range of systems. Of course, performance issues are very dependent on implementation details.

How does Greylisting work?

Greylisting works by leveraging the standard SMTP client design expectation to retry sending mail after an initial 4yz temporary rejection response is issued by the server. When the greylisted recorded SMTP client reschedules and retries the same transaction, the GreyListing server will allow the greylisted recorded sender to continue with the transaction.

While the idea of using an intentional 4yz rejection to force SMTP clients to retry sending mail would naturally be considered a radical concept for the IETF purist and most likely would not have been endorsed as an IETF standard protocol, the proof of concept has long been established as a very effective means to control certain types of malicious and abusive mail senders and today, Greylisting is a widely recognized mail filtering method and Greylisting SMTP Servers are widely implemented by many in the IETF mail community.

What sort of mail senders does Greylisting address?

By leveraging the SMTP retry expectation for clients, Greylisting is very effective against mail senders who anonymously and randomly

perform a "Single Shot" mail sending attempt and will never repeat the same transaction after the sender has been initially rejected. The high payoff has been the nearly 100% of all mail senders behaving as "single shot" mail senders are abusive and/or malicious in nature.

Greylisting can not address abusive mail senders using compliant SMTP mail clients. However, it has been observed that many abusive mail senders will retry again and often immediately within a short time delay. Hence, the Greylisting concept includes the idea of using a "Blocking Time" factor where a greylisted recorded mail sender is blocked for a certain time period. Only when the blocking time has expired, will the GreyListing server finally allow the mail sender to continue with the transaction.

What sort of impact has Greylisting had with Mail Delivery?

Greylisting has been designed since its conception to satisfy certain criteria:

- Enforce SMTP compliance with expected SMTP retry strategies,
- Limit abusive mail senders ability to circumvent the blocking,
- Have minimal impact on users, and
- Require minimal maintenance at both the user and administrator level.

The first immediate impact are the increasing delays in mail delivery due to the wide range in Greylisting blocking time values which can be seconds, minutes to hours. Since SMTP has a standard recommendation to implement a Progressive Retry queuing strategy (see section 4.5.4.1 in RFC5321 [RFC5321]) where the first few attempts have short delays (i.e. two attempts within the first hour) with a progressive back off longer delay before the maximum attempts (i.e. over 4-5 days) are exhausted, there are increasing wasted attempts and foremost higher delays in delivering mail. When a SMTP client implements an initial retry lower than the remote GreyListing Server blocking time, the SMTP client will have increasing wasted attempts overhead. When the SMTP client implements an initial retry delay higher than the remote GreyListing Server blocking time, the SMTP client will have unnecessary wasted mail delays in delivering mail.

With the increasing deployment of Greylisting mail servers, the second impact is such that even the SMTP server who does not employ Greylisting, will more than likely increasingly connect to a remote mail server that does employ Greylisting and will experience the temporary rejection overhead requiring additional mail sending

retries.

The third impact is that many GreyListing servers now use the rejection idea at the connection level using a 421 greeting response which may be a different retry condition than a 45z rejection response issued at the MAIL FROM or DATA state. Since many MTA clients see a 421 as a possible loading limit, it may use this to immediately reschedule a retry using a different MX/IP host..

Overall, Greylisting was designed to address the high abuse of "single shot" anonymous mail senders, however it was done at the expense of legitimate mail senders experiencing wasted mail attempts and increasing delivery delays and with improper GreyListing server and client settings, SMTP clients may now have to revisit their queuing strategies to address the Greylisting overhead related issues.

This specification provides insights into preparing a low impact Greylisting Server by providing some recommendations for blocking delays and defining a formal structure GreyListing server to optionally include a suggested "retry=time-delay" information in the server's 4yz temporary text responses.

This specification does not attempt to alter existing IETF standard SMTP and non-IETF standard Greylisting protocols other than to provide augmented Greylisting techniques to help alleviate the overhead associated with Greylisting in the client/server SMTP transport process. SMTP servers supporting this extension will only be altering 4yz greylisting responses which is out of scope in RFC5321. Greylisting is not part of SMTP and is implemented as an "add-on" component. SMTP clients supporting this extension will only be factoring in a new time factor for their existing retry and queuing method where the exact retry and queuing methods in placed is also out of scope in RFC5321.

The Greylisting method specified in this document is a complementary method to any other existing mail filtering control systems, and is not intended as a replacement for those other methods. In fact, it is expected that abusive mail senders will eventually try to minimize the effectiveness of this method of blocking, and Greylisting is designed to limit options available to the mail senders when attempting to do so. The positive outcome of Greylisting is that the only methods of circumventing it will tend to make other mail filtering control techniques just that much more effective (primarily DNS and other methods of blacklisting based on IP address) even after any adaptation by the abusive mail senders has occurred.

1.2. Document Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.3. Definitions and Acronyms

MTA Mail Transfer Agent. Sender or Receiver of mail. Generally viewed as a router within a MSA intra-network where there is a inherent authentication.

MUA Mail User Agent. Online or offline mail reader/writer software. Typically has its own MTA component for sending mail.

MSA Mail Submission Agent. Generally associated with a MUA sending message to a ISP or ESP where there is an authorized or authenticated association with the MUA.

MDA Mail Destination Agent. Generally associated as the final destination of the message where the message is typically targeted for a local user. If the MDA is going to route the mail, then its behaving more as a MSA or MTA.

1.4. Syntactic Notation

This specification uses the Augmented Backus-Naur Form (ABNF [RFC5234]) notation for the formal definition of the syntax for the "retry=time-delay" hint.

1.5. Out of Scope Considerations

The following are out of scope considerations in this specification:

- o how Greylisting information is recorded in databases,
- o what additional mail information is recorded in databases beyond the Triplet recording, and
- o server reasons for an 4yz response outside a Greylisting reason, such as SMTP Traffic Control concepts.

2. Greylisting Basic Framework

The basic idea of GreyListing is:

1. MTA Client initiates a mail delivery attempt to a remote GreyListing compliant mail receiver (MDA),

2. The GreyListing Server collects first time session information about the sender such as connection IP, MAIL FROM and RCPT TO called the Triplet.

3. If the Triplet was never recorded before, the Triplet is recorded and a 4yz rejection server response with a recommended "retry=time-delay" hint is issued where the time reflects the blocking time the sender can attempt again and proceed with the transaction.

4. If the Triplet was recorded, a check is performed to determine if the blocking time has expired. If not, another 4yz rejection response with a new "retry=time-delay" hint reflecting the new blocking time is issued.

5. When the sender tries again with the same recorded information after the blocking time has expired, then the sender has passed the server's greylist test and is allowed to proceed to send the mail.

One of the essential goals of this specification is to reduce the network and communications overhead in sender attempts and to reduce mail delivery delays by implementing the server "retry=time-delay" hints in the 4yz greylist responses.

2.1. Greylist Recording Parameters

Greylisting Server implementations vary in ways which may include many factors including how senders are traced, accepted, how record expires, the history of sender transactions and including but not limited to how senders are temporarily or permanently white listed. The original Harris [HARRIS] Greylisting specifications offers a range of ideas that are considered.

This specification concentrate on the three principle parameters that make up the fundamental backgound of a Greylist system:

Sender Triplet

Blocking Time Delay

Triplet Expiration Time

2.2. Recording Sender Information (Triplet)

In the classic Greylisting protocol described in HARRIS [HARRIS], a Triplet is the unique combination of connection IP, the reverse address (MAIL FROM) and the forwarding address (RCPT TO) used to track the sender. When the sender retries with the same triplet, a lookup can be perform to determine its Greylist status. However, depending on the Greylist server implementation, it can reject at different points in the SMTP state machine and may not collect the entire triplet information.

While it is out of scope how a SMTP session Triplet is collected and what SMTP session data points it contains, the key point is a specific Triplet used to track the MTA for an initial transaction attempt and subsequent retries in order to control it during the Greylisting Server blocking time.

The following is an implementation example for triplet recording:
Sender-Triplet = triplet-alg(CIP, RPATH, FPATH)

where

CIP	is the connection IP address of the client,
RPATH	is the MAIL FROM reverse-address or domain,
FPATH	is the RCPT TO forwarding address
triplet-alg	is the algorithm used to generate a database tracking key.

One example of a triplet-alg is using a standard hashing algorithm as such SHA1 with BASE64 encoding.

BASE64(SHA1(CIP, RPATH, FPATH))

Other tracking methods such as index keys in SQL database tables are often common with Greylisting server implementations. This specification does not define an formal triple-alg method. Any SMTP data can be used as long as it represents Greylisting servers method for consistent tracking transactions , its initial rejection and subsequent acceptance with expected retries.

2.3. SMTP Server Rejection Points

Greylisting assumes a triplet recording (IP, FROM and TO), however a Greylisting server can reject at any point in the SMTP state machine by recording less information about the sender. This specification hopes to assist the MTA to determine when a temporary rejection is greylist related apart from other reasons which can be a factor in how an MTA client will reschedule new attempts.

2.3.1. Connection Greeting

Many SMTP servers will use a 421 response during the greeting as a way to limit connections and control load.

A GreyListing server deciding to greylist a client at the connection greeting MUST use a 421 reply code and SHOULD include a "retry=time-delay" hint as part of the text response.

The "retry=time-delay" hint will help the MTA decide what sort of rejection was imposed by distinguishing between loading limit or greylist rejection. Without the "retry=time-delay" hint, a MTA can try an alternative MX immediately (without delay) and the rejection may still occur. Including the "retry=time-delay" hint will assist the MTA to better reschedule the retry.

A GreyListing Server rejecting at the connection level is recording only the connection IP to track the sender.

2.3.2. EHLO/HELO

A GreyListing server deciding to greylist a client as a response to the EHLO or HELO command SHOULD use a 451 reply code and SHOULD include a "retry=time-delay" hint as part of the text response. The hint will help the MTA decide when a new attempt should be attempted.

A GreyListing Server rejecting at the EHLO is recording the connection IP and EHLO/HELO machine host name.

Note: The editor has no information on the existence of Greylisting servers that perform a 4yz rejection at the EHLO or HELO command for greylisting reasons.

2.3.3. MAIL FROM

A GreyListing server deciding to greylist a client as a response to the MAIL FROM command SHOULD use a 451 reply code and SHOULD include a "retry=time-delay" hint as part of the text response. The hint will help the MTA decide when a new attempt should be attempted.

A GreyListing Server rejecting at the MAIL FROM is recording the connection IP and MAIL FROM sender address.

2.3.4. RCPT TO

A GreyListing server deciding to greylist a client as a response to the RCPT TO command SHOULD use a 451 reply code and SHOULD include a "retry=time-delay" hint as part of the text response. The hint will help the MTA decide when a new attempt should be attempted.

A GreyListing Server rejecting at the RCPT TO is recording the connection IP, MAIL FROM and RCPT TO addresses.

2.3.5. DATA

A GreyListing server deciding to greylist a client as a response to the DATA End of Data (EOD) SHOULD use a 451 reply code and SHOULD include a "retry=time-delay" hint as part of the text response. The hint will help the MTA decide when a new attempt should be attempted.

Generally, a GreyListing server will allow the DATA command in order to capture the actual RFC5322 [RFC5322] message before a greylist response is issued. The reasons are beyond the scope of this specification.

A GreyListing Server rejecting at the DATA may be recording more information besides the triplet information, i.e. Message-Id header.

2.4. 4yz Format Structure

Many current Greylisting Servers use varying text responses with informal language try again time text information. The following are known forms of existing Greylisting Servers which expose a form of time hints within the text response:

421 This server implements greylisting, please try again in # seconds

450 4.7.1 <RCPT>: Recipient address rejected: Greylisted for # minutes

450 4.7.1 <RCPT>: Recipient address rejected: Greylisted for # seconds

451 4.7.1 Greylisting in action, please come back in HH:MM:SS

451 Greylisted for # seconds

451 Greylisted, please try again in # seconds

451 Greylisting enabled, try again in # minutes

It is possible for existing MTA clients currently supporting the parsing and extraction of the time factor with the known informal responses from existing Greylisting servers and this specification does not attempt to limit specific MTA client implementations which may already exist.

This specification offers a formal structure the Greylisting Server MAY use within their 4yz responses and the MTA client MAY use to detect and extract the retry information consistently without error using a single format within the 4yz response containing the following structured "retry=time-delay" tag:

retry=[DD-]HH:MM:SS

The [DD-]HH:MM:SS part is the time delay the MTA SHOULD wait before attempting to send the mail again. It is not a specific time of day, but rather the amount of GreyListing Server blocking time expected by the server before the client SHOULD try again. An MTA client ignoring this information, attempting again before the blocking time has expired, is a wasted attempt and can delay the mail delivery well beyond the GreyListing server blocking time.

In ABNF [RFC5234], GreyListing server response syntax is:

```
   Reply-Line  = ( Reply-Code [ SP textstring ] CRLF ) /
                 ( Reply-Code "-" [ SP textstring ] CRLF
                   *( Reply-Code "-" [ textstring ] CRLF )
                   Reply-Code [ SP textstring ]  CRLF )

   Reply-Code  = "421" / "450" / "451"

   textstring  = 1*(%x09 / %x20-7E) [SP retryhint [ SP expirehint ] ]

   retryhint   = "retry=" time-delay

   expirehint  = "expire=" time-expire

   time-delay  = ( [days "-"] hours ":" minutes ":" seconds) / totalsecs

   time-expire = ( [days "-"] hours ":" minutes ":" seconds) / totalsecs

   days        = 2DIGIT   ; 00-99

   hours       = 2DIGIT   ; 00-23

   minutes     = 2DIGIT   ; 00-59

   seconds     = 2DIGIT   ; 00-59

   totalsecs   = 6*DIGIT  ; 0-999999
```

   Examples:

Single line responses:

```
  450 4.7.1. Greylist enabled. retry=00:02:00
  451 Temporary rejection. retry=00:00:30
  450 4.7.1. Temporary Greylist rejection. retry=01-00:10:00
  451 TempFail Retry=00:00:55
  421 Your connection is greylisted. Please try again later (retry=00:01:00)
```

Multiple lines response:

```
  451-Greylisted. See policy http://example.com/GreyList-Policy
  451 Retry=00:02:00
```

   For multiple lines responses, the retryhint MUST be provided in the
   last line of the response.

2.4.1. 421 vs 45z Reply Codes

GreyListing Servers may issue 421 or 45z responses at any point in the SMTP session. However, RFC5321 recommends 421 be used at the greeting and for server interruption events. This specification recommends keeping with the SMTP RFC5321 recommendations for 421 and only use 45z for non-Greeting rejections responses. All SMTP compliant MTA will always follow 4yz for scheduling a retry, but the difference is a 421 can trigger an immediate retry attempt without delay at the next MX IP address, if any, where a GreyListing server will most likely reject the new attempt due to the blocking time.

> IMPLEMENTATION NOTE: RFC5321 recommends a specific 450 reply code for temporary rejections related to local policy reasons. HARRIS used 451 to make it distinctive as a greylist response. This specification recommends using 450, however, it is recognized that many existing Greylisting servers already use 451 as the reply code. MTA MUST NOT depend on 450 or 451 to make retry decisions. All 4yz responses MUST be interpreted as a temporary rejection.

When the "retry=time-delay" hint is implemented in the response, compliant MTA will be able to determine the difference between a load restriction and a greylisted rejection to appropriately reschedule a new attempt at the GreyListing server's suggested time hint.

2.5. Recommended Blocking Times

This specification does not impose any specific blocking delay value when 4yz rejections are issued by servers, other than to suggest that timely delivery of mail to users remains to be an inherent expectation by SMTP clients and SMTP servers.

The GreyListing server blocking times vary greatly in practice, but there is empirical evidence a majority of systems use a 1 to 5 minute delay. Many use 10 minutes or 15 minutes. Many use less than 1 minute, like 30 to 55 seconds. The latter tend to be systems who wish to lower impact with immediate and timely mail delivery delays. However, this can be wasteful attempts when the MTA is operating blindly with unknown blocking times imposed by Greylisting Servers.

When it comes to a recommendation, there is no GreyListing logic to suggest that long delays be use when the goal of Greylisting senders is to address the anonymous random "single shot" senders where their triplet will never be the same. Delaying good SMTP senders for extended unreasonable periods defeats the goal of Greylisting.

Since there is no clear recommendation for a blocking time delay (other than to keep it short as possible), this specification offers

the "retry=time-delay" hint as a method to alleviate the uncertainty in the wasted attempts and delays in timely mail delivery.

3. SMTP Service Keyword

GREYLIST is a new ESMTP [RFC1651] service keyword. The GreyListing Server MAY add this optional keyword as a response to EHLO command. EHLO response Format:

250-GREYLIST [server-options]

If the GREYLIST keyword is presented as part of the EHLO response, it means the server has Greylisting implemented and 4yz responses are possible due to a Greylist decision by the server to impose on the client. The keyword is not necessary and the server can still provide 4yz temporary rejections.

The optional server-options provides space separated attributes reflecting the server Greylisting information the server wishes to expose. Currently the following optional attributes are defined:

RETRY means that 4yz responses related to GreyListing will have "retry=time-delay" information. The attribute is optional and not required to issue 4yz responses with "retry=time-delay" hints.

3.1. SMTP Client/Server Implementation

3.1.1. SMTP Server Implementation

The SMTP server MAY add support for the GREYLIST service keyword in the EHLO response. If the SMTP server adds the GREYLIST service keyword without the RETRY attribute, it MAY add the "retry=time-delay" hint to 4yz responses. If the SMTP server adds the GREYLIST service keyword with the RETRY attribute, it MUST add the "retry=time-delay" hint to to 4yz responses.

3.1.2. SMTP Client Implementation

The SMTP client MAY read the GREYLIST service keyword exposed by the EHLO response and it MAY support the usage of the "retry=time-delay" hint in 4yz responses and are not obligated to honor the SMTP servers recommended retry delay.

If the SMTP server offers the GREYLIST keyword with the RETRY attribute, the SMTP client SHOULD consider supporting the usage of the server's recommended retry delay in 4yz responses with "retry=time-delay" hints.

If a SMTP client is rejected by the Greylisting Server during the session, the client SHOULD NOT attempt to start a new transaction during the same session and SHOULD immediately issue a QUIT command to end the session. Its been observed that some mail senders will hold the connection for 1-5 minutes and retry the same mail transaction or a new transaction. The SMTP server rejecting the initial transaction MAY stop accepting any new transactions attempts during the same session.

If a SMTP server offers a "retry=time-delay" hint which results in a wasted 2nd attempt and requires additional attempts, the SMTP client MAY begin to ignore the server's "retry=time-delay" hint after the 2nd wasted retry. The SMTP client implementation can decide what limits to place on honoring "retry=time-delay" hints and wasted attempts it provides.

4. Examples

Example with no extended codes:

```
S: serverdomain.com, welcome ESMTP v2.0
C: EHLO mail.clientdomain.com
S: 250-GREYLIST
S: 250-HELP
C: MAIL FROM:<jqpublic@emaildomain.com>
S: 250 User OK
C: RCPT TO:<localuser@serverdomain.com>
S: 451 Greylisted. Please Disconnect now. retry=00:01:00
C: QUIT
S: 221 Goodbye
```

In the above example, the client can extract the "retry=00:01:00" information and reschedule a 2nd mail delivery attempt at the current attempt time plus 1 minute later and not before. It can reschedule at a later time it if chooses to do so.

Example with extended codes:

```
S: serverdomain.com, welcome ESMTP v2.0
C: EHLO mail.clientdomain.com
S: 250-ENHANCEDSTATUSCODES
S: 250-GREYLIST
S: 250-HELP
C: MAIL FROM:<jqpublic@emaildomain.com>
S: 250 2.1.0 User OK
C: RCPT TO:<localuser@serverdomain.com>
S: 450 4.7.1 Greylisted. Please Disconnect now. retry=00:10:00
expire=02-00:00:00
C: QUIT
S: 221 2.3.0 Goodbye
```

In the above example, the client can extract the "retry=00:05:00" information and reschedule a 2nd mail delivery attempt at the current attempt time plus 10 minutes and not before. It can reschedule at a later time, however since a "expire=time-expire" hint was provide, it should complete the new attempts before the indicated 2 days expiration. If the attempt is done after 2 days, the server will greylist reject the MTA again.

Example of connection rejection:

```
S: 421 4.7.1 Greylist enabled. Try again later. retry=00:10:00
```

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Security Considerations

One possible security concern envisioned is a DoS attack when "retry=time-delay" information is exposed by the GreyListing server where by a malicious sender may attempt to overwhelm the server during the server's retry time exposing a time window when the server has indicated system availability for mail acceptability. However, since security measures to mitigate DoS is a required operational factor, a GreyListing Server will inherently be prepared for DoS attacks with managed loading limits with or without "retry=time-delay" Greylist responses, thus there is no expected technical concern by exposing Greylist "retry=time-delay" hints. With or without this specification, all SMTP servers SHOULD be prepared for DoS attacks of all kinds.

Another arguable security concern is related to the idea a formal SMTP extension can possibly lower the effectiveness of Greylisting when abusive mail senders adapt to the server's suggested retry times. This concern does not seem to have weight since adaptation can occur with or without the extension simply by complying to SMTP retry recommendations. Greylisting remains effective because legacy abusive systems do not adapt. In fact, a "retry=time-delay" hint implementation provides a means to help avoid abusive redundancy and reduced random overloading of connections at unmanaged random times by MTA clients of all flavors. A "retry=time-delay" hint may actually be purposely calculated to provide a time window when there is less loading for legitimate and abusive senders.

7. Acknowledgements

The following individuals contributed input and guidance in the production of this specification:

Claus Assmann, Frank Ellerrman, Tim Kehres, John Klensin, S. Moonesamy, Keith Moore, Ken Raeburn, Paul Smith.

Please note acknowledgement does not imply any specific endorsement of this specification other than they have provided important pros and cons input which helped mold the specification.

8. References

8.1. Normative References

[RFC1651] Klensin, J., Freed, N., Rose, M., Stefferud, E., and D. Crocker, "SMTP Service Extensions", RFC 1651, July 1994.

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

[RFC3463] Vaudreuil, G., "Enhanced Mail System Status Codes", RFC 3463, January 2003.

[RFC4408] Wong, M. and W. Schlitt, "Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1", RFC 4408, April 2006.

[RFC4871] Allman, E., Callas, J., Delany, M., Libbey, M., Fenton, J., and M. Thomas, "DomainKeys Identified Mail (DKIM) Signatures", RFC 4871, May 2007.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.

[RFC5672] Crocker, D., "RFC 4871 DomainKeys Identified Mail (DKIM) Signatures -- Update", RFC 5672, August 2009.

8.2. Informative References

[HARRIS] Harris, E., "The Next Step in the Spam Control War: Greylisting", 2003, <http://projects.puremagic.com/greylisting/whitepaper.html>.

Appendix A. Additional Greylist Parameters

Greylisting Server implementations vary in ways which may include many factors including how senders are traced, accepted, how record expires, the history of sender transactions and including but not limited to how senders are temporarily or permanently white listed. The original Harris [HARRIS] Greylisting specifications offers a range of ideas that are considered. The following are just of a few of additional parameters that are considered by servers:

- Whitelist Record Expiration:

  Whitelist Record Expiration is used to allow a previous greylisted sender a time window where it can be temporarily or permanently whitelisted depending on the implementation. This is a local policy consideration, however, it should be noted that redundant greylisting of a common MTA is not considered reasonable. At some point, the MTA is a trusted source of mail and the MTA SHOULD be permanently whitelisted. The main idea with a temporary whitelisting is that its possible a future transaction can be a compromised user transaction.

- Class C IP Address Tracking:

  Class C IP Address Tracking allows a Greylisting server to control a greylisted MTA who retries using a different class C address. This is typical in larger outbound farms where many machines are used to send mail. If Class C is not considered, MTAs using a

different IP will be unnecessarily rejected after delaying within a blocked time.

Appendix B. Augmenting Other Standard Email Filters Methods

It is possible for a GreyListing server to combine other mail filtering techniques, methods and session information to determine if a sender should be greylisted. While the augmentation of these additional methods is out of the scope, the following are some suggestions that may help minimize a GreyListing Server impact. on MTAs.

SPF SPF (Sender Policy Framework) [RFC4408] can be used to help validate a sender's IP association with the return path domain. A SPF SOFTFAIL or FAIL (if not used for rejection) result could be used to help decide when Greylisting should be employed on the sender. While a PASS result is not a trusted condition, a local policy may use a PASS to skip Greylisting mail checks.

DKIM DKIM (Domain Key Identified Mail) [RFC5672] can be used to help authenticate the transactions from trusted DKIM mail signers. If the signer is considered is trusted source, this can help eliminate the need to greylist the sender.

Appendix C. TO DO LIST

1. Possible section showing real proof of concept examples.

2. Review the SMTP Service Keyword and determine how SHOULD|MAY|MUST is applied.

Authors' Addresses

Hector Santos (editor)
Santronics Software, Inc.
15600 SW 158 ST Suite #306
Homestead, Florida, FL 33033
United States of America

Email: hsantos@santronics.com
URI: http://www.santronics.com

Evan Harris
puremagic.com

Email: eharris@puremagic.com