Individual submission
Internet-Draft
Intended status: Standards Track
Expires: August 25, 2012

M. Kucherawy
Cloudmark
D. Crocker
Brandenburg InternetWorking

February 22, 2012

Email Greylisting: An Applicability Statement for SMTP
draft-ietf-appsawg-greylisting-05

Abstract

This memo describes the art of email greylisting, the practice of providing temporarily degraded service to unknown email clients as an anti-abuse mechanism.

Status of this Memo

This Internet-Draft will expire on August 25, 2012.

Copyright Notice

Table of Contents

1. Introduction

Preferred techniques for handling email abuse explicitly identify good actors and bad actors, giving each significantly differential service. In some cases an actor does not have a known reputation; this can justify providing degraded service, until there is a basis for provider better service. This latter approach is known as "greylisting". Broadly, the term refers to any degradation of service for an unknown or suspect source, over a period of time. The narrow use of the term refers to generation of an SMTP temporary failure reply code for traffic from such sources. There are diverse implementations of this basic concept, and, predictably therefore, some blurred terminology.

This memo documents common greylisting techniques and discusses their benefits and costs. It also defines terminology to enable clear distinction and discussion of these techniques.

There is some confusion in industry that conflates greylisting with an SMTP temporary failure for any reason. The purpose of this memo is also to dispel such confusion.

1.1. Background

For many years, large amounts of spam have been sent through purpose-built software, or "spamware", that supports only a constrained version of SMTP. In particular, such software does not perform retransmission attempts after receiving an SMTP temporary failure. That is, if the spamware cannot deliver a message, it just goes on to the next address in its list since, in spamming, volume counts for far more than reliability. Greylisting exploits this by rejecting mail from unfamiliar sources with a "transient (soft) fail" (4xx) [SMTP] error code. Another application of greylisting is to delay mail from newly seen IP addresses on the theory that, if it's a spam source, then by the time it retries, it will appear in a list of sources to be filtered, and the mail will not be accepted.

Early references for greylisting descriptions and implementations can be found at [SAUCE] and [PUREMAGIC].

1.2. Definitions

1.2.1. Keywords

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].

1.2.2. E-Mail Architecture Terminology

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

2. Types of Greylisting

Greylisting is primarily performed at some phase during an SMTP session. A set of attributes about the client-side SMTP server are used for assessing whether to perform greylisting. At its simplest, the attribute is the IP address of the client and the assessment is whether it has previously connected, recently. More elaborate attribute combinations and more sophisticated assessment, can be performed. The following discussion covers the most common combinations.

2.1. Connection-Level Greylisting

Connection-level greylisting decides whether to accept the (TCP) connection from a "new" [SMTP] client. At this point in the communication between the client and the server, the only information known to the receiving server is the incoming IP address. This, of course, is often (but not always) translatable into a host name.

The typical application of greylisting here is to keep a record of SMTP client IP addresses and/or host names (collectively, "sources") that have been seen. Such a database acts as a cache of known senders and might or might not expire records after some period. If the source is not in the database, or the record of the source has not reached some required minimum age -- such as 30 minutes since the initial connection attempt -- the server does one of the following, inviting a later retry:

- returns a 421 SMTP reply, and closes the connection;

- returns a different 4yz SMTP reply to all further commands in this SMTP session

A useful variant of the basic known/unknown policy is to limit greylisting to those addresses that are on some list of IP addresses known to be affiliated with bad actors. Whereas the simpler policy affects all new connections, including those from good actors, the constrained policy applies greylisting actions only to sites that already have a negative reputation.

2.2. SMTP HELO/EHLO Greylisting

HELO/EHLO greylisting refers to the first [SMTP] command verb in an SMTP session. It includes a single, required parameter that is supposed to contain the client's fully-qualified host name or its literal IP address.

Greylisting implemented at this phase retains a record of sources coupled with HELO/EHLO parameters. It returns 4yz SMTP replies to all commands until the end of the SMTP session if that tuple has not previously been recorded or if the record exists but has not reached some configured minimum age.

2.3. SMTP MAIL Greylisting

MAIL command greylisting refers to the [SMTP] command verb in an SMTP session that initiates a new transaction. It includes at least one required parameter that indicates the return email address (RFC5321.MailFrom) of the message being relayed from the client to the server.

Greylisting implemented at this phase retains a record of sources coupled with return email addresses. It returns 4yz SMTP replies to all commands for the remainder of the SMTP session if that tuple has not previously been recorded or if the record exists but has not met some configured minimum age.

2.4. SMTP RCPT Greylisting

RCPT greylisting refers to the [SMTP] command verb in an SMTP session that specifies intended recipients of an email transaction. It includes at least one required parameter that indicates the email address of an intended recipient of the message being relayed from the client to the server.

Greylisting implemented at this phase retains a record of tuples that combines the provided recipient address with any combination of the following:

- the source, as described above;
- the return email address;
- recipients of the message

If the selected tuple is not found in the database, or if the record is present but has not reached some configured minimum age, the greylisting MTA returns 4yz SMTP replies to all commands for the

remainder of the SMTP session.

Note that often a match on a tuple involving the first valid RCPT is sufficient to identify a retry correctly, and further checks can be omitted.

2.5. SMTP DATA Greylisting

DATA greylisting refers to the [SMTP] command verb in an SMTP session that transmits the actual message content, as opposed to its envelope details (see [MAIL]).

This type of greylisting can be performed at two places in the SMTP sequence:

1. on receipt of the DATA command, because at that point the entire envelope has been received (i.e., all MAIL and RCPT commands have been issued);

2. on completion of the DATA command, i.e., after the "." that terminates transmission of the message body, since at that point a digest or other analysis of the message could be performed.

Some implementations do filtering here because there are clients that don't bother checking SMTP reply codes to commands other than DATA.

Numerous greylisting policies are possible at this point. All of them retain a record of tuples that combine the various parts of the SMTP transaction in some combination, including:

- the source, as described above;

- the return email address;

- the recipients of the message, as a set or individually;

- identifiers in the message, such as the contents of the RFC5322.From or RFC5322.To fields;

- other prominent parts of the content, such as the RFC5322.Subject field;

- a digest of some or all of the message content, as a test for uniqueness;

- analysis of arbitrary portions of the message body.

(The last four items in that list are only possible at the end of

DATA, not on receipt of the DATA command.)

If the selected tuple is not found in the database, or if the record exists but has not reached some configured minimum age, the greylisting MTA returns 4yz SMTP replies to all commands for the remainder of the SMTP session.

2.6. Additional Heuristics

Since greylisting seeks to target spamware, it follows that being able to identify spamware within the SMTP context beyond the simple notion of "not seen before" would be desirable. A more targeted approach might also include in its selection such heuristics as:

- if a [DNSBL] lists an IP address but the implementer wishes to be cautious with mitigation actions rather than blocking traffic from the IP address outright, then subject it to greylisting;

- if the value found in a PTR record follows common naming patterns for dynamic IP addresses, then subject it to greylisting.

2.7. Exceptions

Most greylisting systems provide for an exception mechanism, allowing one to specify IP addresses, IP address [CIDR] blocks, hostnames or domain names that are exempt from greylisting checks and thus whose SMTP client sessions are not subject to such interference.

Likely candidates to be excepted from greylisting include those known not to retry according to a pattern that will be observed as legitimate, and those that send so rarely that they will age out of the database. In both cases the excepted source is known not to be an abusive one by the site implementing greylisting. Otherwise, typical non-abusive senders will enter the exception list on the first proper retry, and remain there permanently.

One could also use a [DNSBL] that lists known good hosts as a greylisting exception set.

3. Benefits and Costs

The most obvious benefit with any of the above techniques is that spamware generally does not retry, and is therefore less likely to succeed, absent a record of a previous delivery attempts.

The most obvious detriment to implementing greylisting is the imposition of delay on legitimate mail. Some popular MTAs do not

retry failed delivery attempts for an hour or more, which can cause expensive delays when delivery of mail is timely. Worse, some legitimate MTAs do not retry at all(!) The counterargument to this "false positive" problem is that email has always been a "best-effort" mechanism, and thus this cost is ultimately low in comparison to the cost of dealing with high volumes of unwanted mail. Still, the actual effect of such delays can be significant, such as altering the tone of a multi-participant discussion to a mailing list.

The cache of information stored about SMTP client history does not benefit legitimate clients that are already listed for acceptance, when the clients are subjected to any kind of reconfiguration, especially such as network renumbering. To the greylisting implementation such clients are once again unknown, and they will once again be subjected to the delay.

Another obvious cost is for the required database. It has to be large enough to keep the necessary history, and fast enough to avoid excessive inefficiencies in the server's operations. The primary consideration is the maximum age of records in the database. If records age out too soon, then hosts that do retry per [SMTP] will be periodically subjected to greylisting even though they are well-behaved; if records age out after too long a period, then eventually spamware that launches a new campaign will not be identified as "unknown" in this manner, and will not be required to retry.

Presuming that known friendly senders will be manually configured as exceptions to the greylisting check, a steady state will eventually be reached wherein the only mail that is delayed is mail from an IP address that has never sent mail before. Experience suggests that the, the vast majority of mail comes from places on such a developed exception list, so after a training period, only a small proportion of mail is actually affected. The training period could be replaced by processing a history of email traffic and adding the IP addresses from which most traffic arrives to the exception list.

Applying greylisting based on actual message content (i.e., post-DATA) is substantially more expensive than any of the other alternatives both in terms of the resources required to accept and temporarily store a complete message body (which can be quite substantial) and any processing that is done on that content. As a consequence, such methods incur more cost during the session and thus is not a typical practice.

4. Unintended Consequences

4.1. Unintended Mail Delivery Failures

There are a few failure modes of greylisting that are worth considering. For example, consider an email message intended for user@example.com. The example.com domain is served by two receiving mail servers, one called mail1.example.com and one called mail2.example.com. On the first delivery attempt, mail1.example.com greylists the client, and thus the client places the message in its outgoing queue for later retry. Later, when a retry is attempted, mail2.example.com is selected for the delivery, either because mail1.example.com is unavailable or because a round-robin [DNS] evaluation produces that result. However, the two example.com hosts do not share greylisting databases, so the second host again denies the attempt. Thus, although example.com has sought to improve its email throughput by having two servers, it has in fact amplified the problem of legitimate mail delay introduced by greylisting.

Similarly, consider a site with multiple outbound MTAs that share a common queue. On a first outbound delivery attempt to example.com, the attempt is grey listed. On a later retry, a different outbound MTA is selected, which means example.com sees a different source, and once again greylisting occurs on the same message.

For systems that do DATA-level greylisting, if any part of the message has changed since the first attempt, the tuple constructed might be different than the one for the first attempt, and the delivery is again greylisted. Some MTAs do reformulate portions of the message at submission time and this can produce visible differences for each attempt.

A host that sends mail to a particular destination infrequently might not remain "known" in the receiving server's database and will therefore be greylisted for a high percentage of mail despite possibly being a legitimate sender.

All of these and other similar cases can cause greylisting to be applied improperly to legitimate MTAs multiple times, leading to long delays in delivery or ultimately the return of the message to its sender. Other side effects include out-of-order delivery of related, sequenced messages.

Address translation technologies such as [NAT] cause distinct MTAs to appear to come from a common IP address. This can cause greylisting to be applied only to the first connection attempt from the shared IP address, meaning future MTAs connecting for the first time will be exempted from the protection greylisting provides.

4.2. Unintended SMTP Client Failures

Atypical SMTP client behaviours also need to be considered when deploying greylisting.

Some clients do not retry messages for very long periods. Popular open source MTAs implement increasing backoff times when messages receive temporary failure messages, and/or degrade queue priority for very large messages. This means greylisting introduces even more delay for MTAs implementing such schemes, and the delay can become large enough to become a nuisance to users.

Some clients do not retry messages at all. This means greylisting will cause outright delivery failure right away for sources, envelopes, or messages that it has not seen before, regardless of the client attempting the delivery, essentially treating legitimate mail and spam the same.

If a greylisting scheme requires a database record to have reached a certain age rather than merely testing for the presence of the record in the database, and the client has a retry schedule that is too aggressive, the client could be subjected to rate limiting by the MTA independent of the restrictions imposed by greylisting.

Some SMTP implementations make the error of treating all error codes as fatal; that is, a 4yz [SMTP] response is treated as if it were a 5yz response, and the message is returned to the sender as undeliverable. This can result in such things as inadvertent removal from mailing lists in response to the perceived rejections.

Some clients encode message-specific details in the address parameter to the [SMTP] MAIL command. If doing so causes the parameter to change between retry attempts, a greylisting implementation could see it as a new delivery rather than a retry, and disallow the delivery. In such cases, the mail will never be delivered, and will be returned to the sender after the retry timeout expires.

A client subjected to greylisting might move to the next host found in the ordered [DNS] MX record set for the destination domain and re-attempt delivery. This can generate an increase in traffic to those alternate servers. Moreover, it is likely that those servers are somehow related and thus will be able to bypass greylisting, either because the servers are in a bypass list at the final destination or due to the traffic their special relationship with the intended recipient implies.

There are some applications that connect to an SMTP server and simulate a transaction up to the point of sending the RCPT command in

an attempt to confirm that an address is valid. Some of these are legitimate applications (e.g., mailing list servers) and others are automated programs that attempt to ascertain valid addresses to which to send spam (a "directory harvesting" attack). Greylisting can interfere with both instances, with harmful effects on the former.

4.3. Address Space Saturation

Greylisting is obviously not a fool-proof solution to avoiding abusive traffic. Bad actors that send mail with just enough frequency to avoid having their records expire will never be caught by this mechanism after the first instance.

Where this is a concern, combining greylisting with some form of reputation service that estimates the likely behaviour for IP addresses that are not intercepted by the greylisting function would be a good choice.

5. Recommendations

The following practices are RECOMMENDED based on collected experience:

1. Implement greylisting based a tuple consisting of (IP address, RFC5321.MailFrom, and the first RFC5321.RcptTo). It has shown sufficient to use only the first RFC5321.RcptTo as legitimate MTAs appear not to reorder recipients between retries. Including RFC5321.MailFrom improves accuracy where the IP address is being matched in clusters (e.g., CIDR blocks) rather than precisely (see below). After a successful retry, allow all further [SMTP] traffic from the IP address in that tuple regardless of envelope information.

2. Include a time window within which a retry from a greylisted host is considered, and ignored otherwise. The default window SHOULD range from one minute to 24 hours. Retries during the period of this window are permitted and satisfy the greylisting test, and thus the client is no longer likely to be spamware; retries after the end of the window SHOULD be considered to be a new message for the purposes of greylisting evaluation (i.e., reset the "first seen" timestamp for that IP address). Some sites use a higher time value for the low end of the window time to match common legitimate MTA retry timeouts, but additional benefit from doing so appears unlikely.

3. Include a timeout for database entries, after which records for IP addresses that have generated no recent traffic are deleted.

This step is intended to re-enable greylisting for an IP address in the event that it has changed "owners", and will subject the client to another round of greylisting. The default SHOULD be at least one week.

4. For an Administrative Domain (ADMD) all inbound border MTAs listed in the [DNS] SHOULD share a common greylisting database and common greylisting policies. This handles sequences in which a client's retry goes to a different server after the first 4yz reply, and it lets all servers share the list of hosts that did retry successfully.

5. To accommodate those senders that have clusters of outgoing mail servers, greylisting servers MAY track CIDR blocks of a size of its own choosing, such as /24, rather than the full IPv4 address. (Note, however, that this heuristic will not work for clusters having machines on different networks.) A similar grouping capability MAY be established based on the domain name of the mail server if one can be determined.

6. Include a manual override capability for adding specific IP addresses or network blocks that always bypass checks. There are legitimate senders that simply don't respond well to greylisting for a variety of reasons, most of which do not conflict with [SMTP]. There are also some highly visible online entities such as email service providers that will be certain to retry, and thus those that are known SHOULD be allowed to bypass the filter.

There is no specific recommendation as to the specific choice of 4yz code to be returned as a result of a greylisting delay. Per [SMTP], however, the only two reasonable choices are 421 if the implementation wishes to terminate the connection immediately, and 450 otherwise. It is possible that some clients treat different 4yz codes differently, but no data are available on whether using 421 versus some other 4yz code is particularly advantageous.

There is also no specific recommendation as to the choice of text to include in the SMTP reply, if any. Some implementers argue that indicating that greylisting is in effect can give spamware a hint as to when to try again for successful delivery, while others suspect that it won't matter to spamware and thus the more likely audience is legitimate senders seeking to understand why their mail is being delayed.

6. Measuring Effectiveness

A few techniques are common when measuring the effectiveness of

greylisting in a particular installation:

- Arrange to log the spam vs. legitimate determinations of messages and what the greylisting decision would have been if enabled; then determine whether there is a correlation (and, of course, whether too much legitimate email would also be affected);

- Continuing from the previous point, query the set of IP addresses subjected to greylisting in any popular [DNSBL] to see if there is a strong correlation.

7. IPv6 Applicability

The descriptions and recommendations presented in this memo are based on many years of experience with greylisting in the IPv4 Internet environment, and so they clearly pertain to IPv4 deployments only.

The greater size of an IPv6 address seems likely to permit differences in behaviours by bad actors, and this could well mean needing to alter the details for applying greylisting; it might even negate any benefits in using greylisting at all. At a minimum, it is likely to call for different specific choices for any greylisting algorithm variables.

In addition, an obvious consideration is that the size of the database required to store records of all of the IP addresses seen will likely be substantially larger in the IPv6 environment.

8. IANA Considerations

No actions are requested of IANA in this memo.

[RFC Editor: Please remove this section prior to publication.]

9. Security Considerations

This section discusses potential security issues related to greylisting.

9.1. Tradeoffs

The discussion above highlights the fact that, although greylisting provides some obvious and valuable defenses, it can introduce unintentional and detrimental consequences for delivery of legitimate mail. Where timely delivery of email is essential, especially for

security-related applications, the possible consequences of such systems need to be carefully considered.

Specific sources can be exempted from greylisting, but of course that means they have elevated privilege in terms of access to the mailboxes on the greylisting system, and malefactors can seek to exploit this.

9.2. Database

The database that has to be maintained as part of any greylisting system will grow as the diversity of its SMTP clients hosts grows, and of course is larger in general depending on the nature of the tuple stored about each delivery attempt. Even with a record aging policy in place, such a database could grow large enough to interfere with the system hosting it, or at least to a point at which greylisting service is degraded. Moreover, an attacker knowing which greylisting scheme is in use could rotate parameters of SMTP clients under its control, in an attempt to inflate the database to the point of denial-of-service.

Implementers could consider configuring an appropriate failure policy so that something locally acceptable happens when the database is attacked or otherwise unavailable.

In practice, this has not appeared as a serious concern, because any reasonable aging policy successfully moderates database growth. It is nevertheless identified here as a consideration as there may be implementations in some environments where this is indeed an issue.

10. References

10.1. Normative References

[EMAIL-ARCH]
Crocker, D., "Internet Mail Architecture", RFC 5598, October 2008.

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

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

10.2. Informative References

[CIDR] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", RFC 4632, August 2006.

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

[DNSBL] Levine, J., "DNS Blacklists and Whitelists", RFC 5782, February 2010.

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

[NAT] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

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

[SAUCE] Jackson, I., "GNU SAUCE", 2001, <http://www.gnu.org/software/sauce>.

Appendix A. Acknowledgments

The author wishes to acknowledge Mike Adkins, Steve Atkins, Mihai Costea, Dave Crocker, Peter J. Holzer, John Levine, Chris Lewis, Jose-Marcio Martins da Cruz, John Klensin, S. Moonesamy, Suresh Ramasubramanian, Mark Risher, Jordan Rosenwald, Gregory Shapiro, Joe Sniderman, Roland Turner, and Michael Wise for their contributions to this memo. The various participants of the MAAWG Open Sessions about greylisting were also valued contributors.

Authors' Addresses

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

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

D. Crocker
Brandenburg InternetWorking
675 Spruce Dr.
Sunnyvale 94086
USA

Phone: +1.408.246.8253
Email: dcrocker@bbiw.net
URI: http://bbiw.net