IETF Emailcore Interim

21 January 2022

Chairs:
Alexey Melnikov <alexey.melnikov@isode.com>
Todd Herr <todd.herr@valimail.com>

Note Well

- This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

- As a reminder:

  - By participating in the IETF, you agree to follow IETF processes and policies.

  - If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.

  - As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.

  - Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.

  - As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Note Well (continued)

- Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:
  - BCP 9 (Internet Standards Process)
  - BCP 25 (Working Group processes)
  - BCP 25 (Anti-Harassment Procedures)
  - BCP 54 (Code of Conduct)
  - BCP 78 (Copyright)
  - BCP 79 (Patents, Participation)
  - https://www.ietf.org/privacy-policy/ (Privacy Policy)

IETF Code Of Conduct Guidelines RFC 7154

- Treat colleagues with respect
- Speak slowly and limit the use of slang
- Dispute ideas by using reasoned argument
- Use best engineering judgment
- Find the best solution for the whole Internet
- Contribute to the ongoing work of the group and the IETF

Administrivia

- This Zoom session is being recorded
- Zoom:
  - https://us06web.zoom.us/j/89359071984?pwd=ZXZSOVBtc0RPWUl1RjhUUGlRZzZQQT09
- Jabber room (discussions/back channel):
  - emailcore@jabber.ietf.org
- Shared note taking:
  - https://notes.ietf.org/notes-emailcore-interim-jan-2022
- ***Note taker***?

Agenda

- Agenda bashing, administrivia, note well (chairs) - 5 mins
- #17 (Deprecated Source Routes) <https://trac.ietf.org/trac/emailcore/ticket/17>
- #9 (G.7.3. Definition of domain name in Section 2.3.5) <https://trac.ietf.org/trac/emailcore/ticket/9>
- #4 (Exploders seem to be prohibited from adding List-* header fields) <https://trac.ietf.org/trac/emailcore/ticket/4>
- #1 (G.1 IP address literals in EHLO) <https://trac.ietf.org/trac/emailcore/ticket/1>
- #47 (Accepting Messages based on EHLO Argument) <https://trac.ietf.org/trac/emailcore/ticket/47>
- #54 (G.7.17 Hop-by-hop Authentication and/or Encryption) <https://trac.ietf.org/trac/emailcore/ticket/54>
- #16 (Review Timeout Specifications) <https://trac.ietf.org/trac/emailcore/ticket/16>
- #12 (G.7.5. Improve description/definition of mailing lists, aliases, and forwarding) <https://trac.ietf.org/trac/emailcore/ticket/12>
- #3 (G.3. Meaning of "MTA" and Related Terminolog) <https://trac.ietf.org/trac/emailcore/ticket/3>

RFC 5321

G.7.10. Further clarifications needed to deprecated source routes?

https://trac.ietf.org/trac/emailcore/ticket/17

Background: RFC 5321 says that source routes are deprecated since 1989, yet at the same time servers must accept them and there are various SHOULDs about whether they can be ignored or rejected by servers, and about when clients can generate them. It also talks about using source routing to work around temporary DNS problems and for mail system debugging.

Agreement on how to deal with this: **strip the document of all mentioning of handling of source routes in text and ABNF, other than to specify their historical use in RFC 821 and point to RFC 821 for implementations that want to implement them for backward compatibility.**

Few minor remaining issues on the following slides.

RFC 5321

G.7.10. Further clarifications needed to deprecated source routes?

F.2. Source Routing

RFC 821 utilized the concept of explicit source routing to get mail from one host to another via a series of relays. Source routes could appear in either the <forward-path> or <reverse-path> to show the hosts through which mail would be routed to reach the destination. The requirement to utilize source routes in regular mail traffic was eliminated by the introduction of the domain name system "MX" record by RFC 974 in early 1986 and the last significant justification for them was eliminated by the introduction, in RFC 1123, of a clear requirement that addresses following an "@" must all be fully-qualified domain names. Issues involving local aliases for mailboxes were addressed by the introduction of a separate specification for mail submission [41]. Consequently, there are no remaining justifications for the use of source routes other than support for very old SMTP clients. Even use in mail system debugging is unlikely to work because almost all contemporary systems either ignore or reject them.

RFC 5321

G.7.10. Further clarifications needed to deprecated source routes?

F.2. Source Routing

Historically, for relay purposes, the forward-path may have been a source route of the form "@ONE,@TWO:JOE@THREE", where ONE, TWO, and THREE MUST be fully-qualified domain names. This form was used to emphasize the distinction between an address and a route. The mailbox (here, JOE@THREE) is an absolute address, and the route is information about how to get there. The two concepts should not be confused.

SMTP servers ***SHOULD*** continue to accept source route syntax as specified in this appendix. If they do so, they SHOULD ignore the routes and utilize only the target domain in the address. If they do utilize the source route, the message MUST be sent to the first domain shown in the address. In particular, a server MUST NOT guess at shortcuts within the source route. SMTP clients ***SHOULD NOT*** attempt to utilize explicit source routing.

RFC 5321

G.7.10. Further clarifications needed to deprecated source routes?

F.2. Source Routing

If source routes appear in mail received by an SMTP server contrary to the requirements and recommendations in this specification, RFC 821 and the text below should be consulted for the mechanisms for constructing and updating the forward-path. A server that is reached by means of a source route (e.g., its domain name appears first in the list in the forward-path) MUST remove its domain name from any forward-paths in which that domain name appears before forwarding the message and MAY remove all other source routing information. Any source route information in the reverse-path SHOULD be removed by servers conforming to this specification.

RFC 5321

G.7.10. Further clarifications needed to deprecated source routes?

F.2. Source Routing

The following information is provided for historical information only, so that the source route syntax and application can be understood if needed.

Syntax:
The original form of the <Path> production in Section 4.1.2 was:

Path = "<" [A-d-l ":"] Mailbox ">"

A-d-l = At-domain *("," At-domain)

At-domain = "@" Domain

For example, suppose that a delivery service notification must be sent for a message that arrived with:
MAIL FROM:<@a.example,@b.example:user@d.example>
The notification message MUST be sent using:
RCPT TO:<user@d.example>

RFC 5321

G.7.3. Definition of domain name in Section 2.3.5

https://trac.ietf.org/trac/emailcore/ticket/9

2.3.5. Domain Names

Paragraph 2:

> The domain name, as described in this document and in RFC 1035 [4], MUST be the entire, fully-qualified name (often referred to as an "FQDN"). Other than an address literal (see Section 4.1.3) where those are permitted, any string that is not a domain name in FQDN form is no more than a reference to be interpreted locally. Such local references for domain names MUST NOT appear in any SMTP transaction (Cf. Section 5). ***Mechanisms for inferring FQDNs from local references (including partial names or local aliases) are outside of this specification and normally the province of message submission. Due to a history of problems, SMTP servers used for initial submission of messages SHOULD NOT make such inferences (Message Submission Servers [41] have somewhat more flexibility) and intermediate (relay) SMTP servers MUST NOT make them.***

John K: The sentence starting with Mechanisms" and the one immediately following it above moved from Section 5.1, but perhaps they should be dropped entirely and/or elaborated on in the A/S.

RFC 5321

G.7.3. Definition of domain name in Section 2.3.5

https://trac.ietf.org/trac/emailcore/ticket/9

2.3.5. Domain Names

Paragraph 3:

> When domain names are used in SMTP, and unless further restricted in this document, names that can be resolved to MX RRs or address (i.e., A or AAAA) RRs (as discussed in Section 5) are permitted, as are CNAME RRs whose targets can be resolved, in turn, to MX or address RRs. There are two exceptions to the rule requiring FQDNs:
>
> * The domain name given in the EHLO command MUST be either a primary host name (a domain name that resolves to an address RR) or, if the host has no name, an address literal, as described in Section 4.1.3 and discussed further in the EHLO discussion of Section 4.1.4.
>
> * The reserved mailbox name "postmaster" may be used in a RCPT command without domain qualification (see Section 4.1.1.3) and MUST be accepted if so used.

The above doesn't require domain names to be "resolvable" anymore, but it still talks about DNS. Does the discussion of MX/A/AAAA belong to this section? Is Section 5 a better place?

RFC 5321

Exploders seem to be prohibited from adding List-* header fields

https://trac.ietf.org/trac/emailcore/ticket/4

3.4.2. Aliases and Mailing Lists

An SMTP-capable host SHOULD support both the alias and the list models of address expansion for multiple delivery. When a message is delivered or forwarded to each address of an expanded list form, the return address in the envelope ("MAIL FROM:") MUST be changed to be the address of a person or other entity who administers the list. However, in this case, ***the message header section (RFC 5322 [12]) MUST be left unchanged***; in particular, the "From" field of the header section is unaffected.

Problem: "MUST be left unchanged" seems to prohibit addition of header fields. Also some mailing lists add tags to Subject header fields. And DMARC workaround strategies result in modified From.

Proposal (replace the last 2 sentences with):

When a message is delivered or forwarded to each address of an expanded list form, the return address in the envelope ("MAIL FROM:") MUST be changed to be the address of a person or other entity who administers the list. **This change to MAIL FROM doesn't affect the header section of the message**.

Ticket #1 - G.1 IP address literals in EHLO

- https://trac.ietf.org/trac/emailcore/ticket/1
- Discussion at IETF 110 landed on leaving text alone in 5321bis, but recommend against use on the public internet in A/S - https://notes.ietf.org/notes-ietf-110-emailcore
- Suggested text for A/S:

2.4 Usage of IP Address in Either EHLO or MAIL FROM Command

If an SMTP client presents an IP address or ‘localhost’ as the argument to the EHLO command for a transaction occurring on the public internet, the SMTP server may refuse any mail from the client as part of established anti-abuse practice. Similar results are likely if the Domain part of the argument to the MAIL FROM command is an IP address literal or ‘localhost’. Experience shows that both are indications of at best a poorly-configured MTA, and at worst a compromised host that’s intentionally configured to hide its identity.

Ticket #47 - Accepting Messages based on EHLO Argument

- https://trac.ietf.org/trac/emailcore/ticket/47
- Duplicate of https://trac.ietf.org/trac/emailcore/ticket/19 and closed as such
- Applicability statement (draft-ieft-emailcore-as-03) already contains this text:

2.1. Handling of the Domain Argument to the EHLO Command

If the Domain argument to the EHLO command does not have an address record in the DNS that matches the IP address of the client, the SMTP server may refuse any mail from the client as part of established anti-abuse practice. Operational experience has demonstrated that the lack of a matching address record for the the domain name argument is at best an indication of a poorly-configured MTA, and at worst that of an abusive host.

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption

- https://trac.ietf.org/trac/emailcore/ticket/54
- Ticket initially asked if 5321bis should discuss either topic.
- Chairs suggested that any mention of either topic should be in the A/S
- Initial draft of suggested text follows

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption (cont'd)

5. Hop-by-hop Authentication and Its Implications (Suggested Text)

Two protocols exist to allow for authentication of different identities associated with an email message - SPF [RFC7208] and DKIM [RFC6376]. A third protocol, DMARC [RFC7489], relies on SPF and DKIM to allow for validation of the domain in the visible From header, and a fourth, ARC [RFC8617], provides a way for each hop to record results of authentication checks performed at that hop.

All of these are outside the scope of this document, but users and implementers of SMTP should be aware of them and most critically of the fact that both SPF and DKIM verification checks can produce different results when a message transits multiple hops vice when it goes directly from the sender to the destination, specifically "FAIL" results. These unanticipated failures can and do affect DMARC verification results on some messages, and so domain owners should be aware of this when setting DMARC policy for their domains. As for message receivers, the ARC protocol is one attempt to provide evidence of the results of previous authentication checks, and it's up to the receiver to decide if they trust the results and how those results might impact message handling at their site.

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption (cont'd)

6. Message Encryption and Its Implications (Suggested Text)

The default method for transmitting text using the SMTP protocol is to do so "in the clear". Years of operational experience have shown that such transmission exposes the message to easy compromise. To mitigate this risk, solutions have been developed to encrypt the message in transit, either just between servers or through the entire path from sender to message store. This section will touch on these methods and the implications of their use.

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption (cont'd)

6.1 Opportunistic Encryption (suggested text)

The most common implementation of message encryption is what's known as "opportunistic encryption". With this method, an SMTP server announces in its greeting that it is capable of supporting TLS encryption through the presence of the "STARTTLS" keyword. The SMTP client then attempts to negotiate an encrypted connection, and if successful, transmits the message in encrypted form; there is no guarantee that the message will be stored in encrypted fashion at its destination, and in fact, storage in plain text should be expected. If negotiation fails, the client falls back to sending the message in clear text.

Most modern implementations of SMTP support this method, and so the vast majority of email traffic is encrypted during its time transiting from the client to the server.

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption (cont'd)

6.2 Required Encryption (suggested text)

Two protocols exist that move server-to-server encryption beyond "nice to have" to "required" - MTA-STS [RFC8461] and DANE for SMTP [RFC7672]. While they differ in their implementation details, SMTP servers relying on either protocol are stating that they only accept mail if the transmission is encrypted with TLS, and a failure to negotiate a secure connection MUST result in the SMTP client refusing to transmit the message. Support for both protocols is widening, but is not yet mandatory.

Ticket #54 - G.7.17 Hop-by-hop Authentication and/or Encryption (cont'd)

6.3 Personal Encryption (suggested text)

The more sophisticated among the end users of SMTP may take advantage of various third party solutions that are designed to encrypt their message immediately upon leaving the MUA, and to do so in such a way that only the individual message recipient(s) can decrypt the message. The various solutions are too numerous to list here, and debugging any issues that may arise in message transmission is likely to be beyond the scope of any MTA administrator's work, but they're nonetheless mentioned here for the sake of completeness.

Ticket #16 - Review Timeout Specifications

- https://trac.ietf.org/trac/emailcore/ticket/16
- Attempted on-list discussion to see if timeouts need to be revisited on the theory that they might be longer than necessary.
- Consensus was to leave them as is, but mention them in the A/S
- Proposed text for A/S

7. Timeout Specifications

The current SMTP specification (5321bis), as well as the two before it (RFC5321 and RFC2821) contain recommendations for minimum timeout values for various operations. Given the age of RFC 2821 (published in April 2001) and advances in networking and computing technology since then, it might seem that these timeout values are far too generous. However, those same advances have been relied by MTA implementers to add more features and functionality to SMTP transactions.

The recommended minimum timeouts are specified as SHOULD, not MUST, and so implementers may choose to use shorter timeouts if operational needs or experience indicate that to be appropriate.

RFC 5321

G.7.5. Improve description/definition of mailing lists, aliases, and forwarding

https://trac.ietf.org/trac/emailcore/ticket/12

The next few slides display current text about mailing lists and aliases. When discussing them, please consider the following question: **is the current definition broken or is it good enough**?

- clarifications and/or adding extra examples is fine
- the bar for changing the definition completely is high and need to have strong WG consensus

RFC 5321

G.7.5. Improve description/definition of mailing lists, aliases, and forwarding

https://trac.ietf.org/trac/emailcore/ticket/12

3.4.2. Aliases and Mailing Lists

2nd paragraph:

> An important mail facility is a mechanism for multi-destination delivery of a single message, by transforming (or "expanding" or "exploding") a pseudo-mailbox address into a list of destination mailbox addresses. When a message is sent to such a pseudo-mailbox (sometimes called an "exploder"), copies are forwarded or redistributed to each mailbox in the expanded list. Servers SHOULD simply utilize the addresses on the list; application of heuristics or other matching rules to eliminate some addresses, such as that of the originator, is strongly discouraged. We classify such a pseudo-mailbox as an "alias" or a "list", depending upon the expansion rules.

RFC 5321

G.7.5. Improve description/definition of mailing lists, aliases, and forwarding

https://trac.ietf.org/trac/emailcore/ticket/12

3.4.2.1. Simple Aliases

> To expand an alias, the recipient mailer simply replaces the pseudo-mailbox address in the envelope with each of the expanded addresses in turn; the rest of the envelope and the message body are left unchanged. The message is then delivered or forwarded to each expanded address.

Note forwarding as an email address portability issue? If we do, is this something for A/S? Or just an example here?

Suggestion to add an example explaining how this works.

Suggestion to do no further changes.

RFC 5321

G.3. Meaning of "MTA" and Related Terminology

https://trac.ietf.org/trac/emailcore/ticket/3

G.3. Meaning of "MTA" and Related Terminology

A terminology issue has come up about what the term "**MTA**" actually refers to, a question that became at least slightly more complicated when we formalized RFC 6409 Submission Servers. Does the document need to be adjusted to be more clear about this topic? Note that the answer may interact with the question asked in Section 2 above. Possibly along the same lines, RFC 2821 changed the RFC 821 terminology from "**sender-SMTP**" and "**receiver-SMTP**" to "**SMTP client**" and "**SMTP server**" respectively. As things have evolved, it is possible that newer terminology is a source of confusion and that the terminology should be changed back, something that also needs discussion.

Question 1: "**sender-SMTP**" and "**receiver-SMTP**" versa "**SMTP client**" and "**SMTP server**". Proposal: no change.

Question 2: definition of MTA (next slide)

RFC 5321

G.3. Meaning of "MTA" and Related Terminology

https://trac.ietf.org/trac/emailcore/ticket/3

2.3.3. Mail Agents and Message Stores

Additional mail system terminology became common after RFC 821 was published and, where convenient, is used in this specification. In particular, SMTP servers and clients provide a mail transport service and therefore act as "**Mail Transfer Agents**" (MTAs). "**Mail User Agents**" (MUAs or UAs) are normally thought of as the sources and targets of mail. At the source, an MUA might collect mail to be transmitted from a user and hand it off to an MTA or, more commonly in recent years, a specialized variation on an MTA called a "**Submission Server**" (MSA) [42]. . At the other end of the process, the final ("delivery") MTA would be thought of as handing the mail off to an MUA (or at least transferring responsibility to it, e.g., by depositing the message in a "message store"). However, while these terms are used with at least the appearance of great precision in other environments, the implied boundaries between MUAs and MTAs often do not accurately match common, and conforming, practices with Internet mail. Hence, the reader should be cautious about inferring the strong relationships and responsibilities that might be implied if these terms were used elsewhere

Proposal: no change, unless the above text is broken.

Done for today

Don't forget to preserve Zoom chat for posterity!