Lemonade
Internet-Draft
Updates: 2554 (if approved)
Intended status: Standards Track
Expires: October 19, 2007

T. Finch
University of Cambridge

April 17, 2007

The QUICKSTART SMTP service extension (simple profile)
draft-fanf-smtp-quickstart-a

Status of this Memo

Copyright Notice

Abstract

This memo specifies modifications to SMTP's start-up sequence which reduce the number of round trips between the client and the server before message transmission starts. This can significantly reduce the delay for interactive message submission over slow links. (This is the simple version of the proposal.)

Document revision

$Cambridge: hermes/doc/qsmtp/draft-fanf-smtp-quickstart-a.xml,v 1.43 2007/04/17 06:46:21 fanf2 Exp $

Table of Contents

1. Introduction

SMTP can require many round trips between the client and server before message transmission starts. This is a particular problem for message submission over slow links, where the resulting delay can easily be two seconds or more. The QUICKSTART SMTP service extension introduces some changes which eliminate round trips and therefore reduce the delay before messages can be transmitted. It is based on the PIPELINING service extension [RFC2920]. The key features of the extension are:

- Earlier announcement of the list of SMTP service extensions supported by the server.
- The QHLO and QTLS commands, which are quicker replacements for the EHLO and STARTTLS commands.
- The ability to pipeline the AUTH command.

In a normal message sumbission connection, the client sends the first MAIL command in its 9th packet; with QUICKSTART the first MAIL command is in the client's 5th packet. Appendix A illustrates these counts in more detail.

Although SMTP QUICKSTART is probably most useful in conjunction with message submission [RFC4409], it MAY also be used with MTA-to-MTA SMTP - though attention should be paid to the security considerations in Section 9.

1.1. Procedural Rubric

Comments and discussion about this draft should be directed to the <lemonade@ietf.org> mailing list - the working group dealing with enhancements to Internet email to support diverse service environments.

This draft should be read in conjunction with [quickstart-b] which describes an alternative profile of these ideas. Draft -A is simpler to implement, whereas draft -B uses extra state to save even more round trips. Draft -A introduces a QTLS command whereas draft -B is generalised to work with security layers set up using the existing STARTTLS and AUTH commands.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this

document are to be interpreted as described in [RFC2119].

The metalinguistic notation used in this memo corresponds to the "Augmented Backus-Naur Form" used in other Internet mail system memos. The reader who is not familiar with that syntax should consult the ABNF specification [RFC4234]. Rules not defined in this specification are either defined in the ABNF core rules or in [RFC2821]. Metalanguage terms used in running text are surrounded by pointed brackets (e.g., <qhlo-id>) for clarity.

3. QUICKSTART SMTP service extension definition

The QUICKSTART SMTP service extension is defined as follows:

- The name of the service extension is "QUICKSTART".

- The EHLO keyword value associated with the extension is "QUICKSTART".

- The extension defines two additional commands: "QHLO" (in Section 5) and "QTLS" (in Section 6).

- Changes to the AUTH command (updating [RFC2554]) are described in Section 7.

- There are no additional parameters to the MAIL or RCPT commands defined by this extension and their maximum lengths are not increased.

4. Extended server greeting

An SMTP server that supports QUICKSTART MUST respond to the client’s connection with an extended greeting according to the syntax below. This greeting has a similar format to the SMTP EHLO reply, that is, the first line is the same as an un-extended greeting, and subsequent lines list the SMTP service extensions supported by the server. The list MUST include the QUICKSTART keyword and the PIPELINING keyword [RFC2920]. The server SHALL give exactly the same list in its greeting as it does in reply to the EHLO command.

Syntax:

```
     qsmtp-greet  =      "220-" Domain [ SP ehlo-greet ] CRLF
                      *( "220-" ehlo-line              CRLF  )
                         "220 " ehlo-line              CRLF

     ehlo-line    =/ "QUICKSTART"
```

The extended greeting allows a client to find out the server’s supported extensions one round trip earlier than in normal SMTP.

5. The QHLO command

An SMTP client that wishes to use QUICKSTART MUST first ensure that the server supports it by checking the server’s greeting against the syntax specified in Section 4. If the client does not want to use TLS, it then issues QHLO as its first command instead of EHLO.

Syntax:

```
     qhlo          =  "QHLO" SP Domain CRLF

     qhlo-ok-rsp   =  *(  "250-" [    text ] CRLF  )
                          "250"  [ SP text ] CRLF
```

The QHLO command has the same semantics as the EHLO command, except that it MAY be pipelined, and the server’s 250 reply to QHLO SHALL NOT include its list of supported extensions nor an enhanced status code [RFC2034] [RFC3463].

The pipelined QHLO command allows a client to save one round trip by not having to wait for the server’s reply before issuing further commands.

6. The QTLS command

An SMTP client that wants to make a QUICKSTART connection using TLS MUST first ensure that the server supports it by checking the server’s greeting against the syntax specified in Section 4. It then issues QTLS as its first command instead of EHLO or QHLO. (The reason for omitting QHLO before QTLS is so that the client does not have to handle the server’s reply to QHLO after sending its TLS client hello.)

Syntax:

```
     qtls  =  "QTLS" CRLF
```

The QTLS command has no parameters. After issuing the QTLS command, the client immediately starts the TLS handshake. The server SHALL NOT issue an SMTP reply to the QTLS command before the TLS handshake. Therefore it is not possible for the server to report problems (as with the 454 reply to the STARTTLS command) except by closing the connection - which it SHOULD NOT do. Instead, the server SHOULD ensure that it can support TLS (e.g. by initializing its TLS subsystem) before listing the extensions it supports, rather than lazily in reaction to a QTLS or STARTTLS command.

After the TLS handshake has been completed, the SMTP protocol is reset to the state at the start of a connection, so the server SHALL issue a QUICKSTART extended 220 greeting <qsmtp-greet>. The server SHALL give exactly the same list in this greeting as it does in reply to a EHLO command issued after a successful STARTTLS. The client SHOULD issue a QHLO command when it has processed the greeting.

If the server requires that the client performs a TLS negotiation before it accepts any commands, then it SHOULD accept the QHLO and QTLS commands before TLS negotiation, as well as the NOOP, EHLO, STARTTLS, and QUIT commands specified in [RFC3207].

Apart from these differences, the QTLS command is the same as the STARTTLS command, and the other requirements specified in [RFC3207] apply, such as discarding state obtained before the TLS handshake.

The QTLS command saves one round trip before the TLS handshake by omitting the server’s reply, and saves another after the TLS handshake by announcing the list of extensions supported by the server earlier than with STARTTLS.

7. Changes to the AUTH command

If the server supports QUICKSTART and AUTH [RFC2554], and the client uses a SASL mechanism which can be completed in one round trip (such as EXTERNAL [RFC4422] or PLAIN [RFC4616]) then the client MAY issue the the AUTH command at any point in a pipelined group. If the authentication fails, the server SHOULD reject all subsequent commands other than AUTH, NOOP, HELO, EHLO, QHLO, or QUIT with a "530 Authentication failure" reply (because the client expects these commands to be interpreted following a successful authentication). If the server supports ENHANCEDSTATUSCODES [RFC2034] [RFC3463], the status code to be returned SHOULD be 5.7.0.

This change allows one round trip to be saved. It is also suggested in [RFC4468].

8. IANA Considerations

This memo defines a new SMTP service extension keyword, "QUICKSTART" in Section 3.

IANA maintains a registry of "WITH protocol types" for use in the "with" clause of the Received header trace fields in an Internet message. Most of the contents of this registry are set out in [RFC3848]. This specification updates the registry as follows:

- The new keyword "QSMTP" indicates that the client used the QUICKSTART extension, that is, it used the QHLO command to start the SMTP session instead of EHLO.

- The new keyword "QSMTPA" indicates that the client used QUICKSTART with the SMTP AUTH [RFC2554] extension, and that it authenticated successfully.

- The new keyword "QSMTPS" indicates that the client used QUICKSTART, and that it successfully completed TLS negotiation to provide a strong transport encryption layer by using the QTLS or STARTTLS commands.

- The new keyword "QSMTPSA" indicates that the client used QUICKSTART and both TLS and AUTH were successfully negotiated (the combination of QSMTPS and QSMTPA).

9. Security considerations

Like the EHLO command, the QHLO command includes a client host name parameter for logging and tracing purposes. This can useful to distinguish different clients behind a NAT, for example. However clients commonly state an incorrect host name, so it SHOULD NOT be relied on. SMTP servers SHOULD use all available client identifiers for logging and tracing, such as its IP address, reverse DNS, TLS certificate, and SMTP AUTH credentials. This memo allows a client to start a TLS handshake without issuing an EHLO or QHLO command, in which case the client host name will not be available at that point in the SMTP conversaion to log in case of any failures.

Apart from that, this memo does not specify any modifications to SMTP that affect security.

10. References

10.1. Normative references

[RFC2034] Freed, N., "SMTP Service Extension for Returning Enhanced Error Codes", RFC 2034, October 1996.

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

[RFC2554] Myers, J., "SMTP Service Extension for Authentication", RFC 2554, March 1999.

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

[RFC2920] Freed, N., "SMTP Service Extension for Command Pipelining", STD 60, RFC 2920, September 2000.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, February 2002.

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

[RFC3848] Newman, C., "ESMTP and LMTP Transmission Types Registration", RFC 3848, July 2004.

[RFC4234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005.

10.2. Informative references

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC4346] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1", RFC 4346, April 2006.

[RFC4409] Gellens, R. and J. Klensin, "Message Submission for Mail", RFC 4409, April 2006.

[RFC4422] Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)", RFC 4422, June 2006.

[RFC4468] Newman, C., "Message Submission BURL Extension", RFC 4468, May 2006.

[RFC4616] Zeilenga, K., "The PLAIN Simple Authentication and Security Layer (SASL) Mechanism", RFC 4616, August 2006.

[old−quickstart−01]
Finch, T., "The QUICKSTART SMTP service extension", Internet Draft draft−fanf−smtp−quickstart−01.txt, February 2007.

[quickstart−b]
Finch, T., "The QUICKSTART SMTP service extension (full profile)", Internet Draft draft−fanf−smtp−quickstart−b−00.txt, April 2007.

Appendix A. Example round trip summaries

In the following examples, a line prefixed with C: indicates the start of a packet from the client to the server, and S: indicates one from the server to the client. Subsequent lines without a prefix are pipelined into the same packet. The C: or S: may be followed by a protocol name - TCP, SMTP or TLS - which indicates a change in the top-most protocol active in the connection. The rest of the line describes a protocol element.

The following figure outlines the sequence of packets at the start of a non-QUICKSTART message submission [RFC4409], including the TCP handshake [RFC0793], starting ESMTP [RFC2821], securing it using STARTTLS [RFC3207] [RFC4346], authenticating [RFC2554] [RFC4422] [RFC4616], and starting to send the message using pipelining [RFC2920]. The MAIL command appears in the 9th packet from the client, and pipelining is not possible before this point.

```
C: TCP  SYN
S:      SYN/ACK
C:      ACK
S: SMTP <greeting>
C:      EHLO <Domain>
S:      <ehlo-ok-rsp>
C:      STARTTLS
S:      220 OK
C: TLS  client hello
S:      server hello, certificate, etc.
C:      key exchange, change cipher, finish handshake
S:      change cipher, finish handshake
C: SMTP EHLO <Domain>
S:      <ehlo-ok-rsp>
C:      AUTH PLAIN <creds>
S:      235 OK
C:      MAIL FROM:<...>
        RCPT TO:<...>
        ...
```

The following shows the use of the QTLS and QHLO commands and pipelined AUTH, to illustrate a QUICKSTART connection. The MAIL command is in the client’s 5th packet.

```
C: TCP  SYN
S:      SYN/ACK
C:      ACK
S: SMTP <qsmtp-greet>
C:      QTLS
   TLS  client hello
S:      server hello, certificate, etc.
C:      key exchange, change cipher, finish handshake
S:      change cipher, finish handshake
   SMTP <qsmtp-greet>
C:      QHLO <Domain>
        AUTH PLAIN <creds>
        MAIL FROM:<...>
        RCPT TO:<...>
        ...
S:      250 OK
        235 OK
        250 OK
        ...
```

Note that the client omitted the QHLO at the start of the previous connection. This avoids the following situation where it has to switch back and forth between SMTP and TLS at the start of the TLS handshake.

```
        ...
S: SMTP <qsmtp-greet>
C:      QHLO <Domain>
        QTLS
   TLS  client hello
S: SMTP 250 OK
   TLS  server hello, certificate, etc.
        ...
```

Appendix B. Changes from previous versions

This memo is derived from [old-quickstart-01] by separating it into two drafts, this one describing the simple profile of the ideas, and [quickstart-b] describing the more advanced profile.

The QUICKSTART extension no longer has any optional features.

PIPELINING is now a required prerequisite for QUICKSTART.

Clarify why commands are rejected after a failed AUTH.

Require consistency between the server’s various lists of supported service extensions.

Some ABNF fixes.

Better consistency with existing SMTP reply codes.

This profile omits the pipelined STARTTLS command, and the extended form of the QHLO command.

Author’s Address

Tony Finch
University of Cambridge Computing Service
New Museums Site
Pembroke Street
Cambridge CB2 3QH
ENGLAND

Phone: +44 797 040 1426
Email: dot@dotat.at
URI: http://dotat.at/

Full Copyright Statement

THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on−line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf−ipr@ietf.org.

Acknowledgment

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