

July 1, 2000

Message Tracking Query Protocol

<[draft-ietf-msgtrk-mtqp-00.txt](#)>

Authors' version: 1.3

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This memo and its companions are discussed on the MSGTRK working group mailing list, ietf-msgtrk@imc.org. To subscribe, send a message with the word "subscribe" in the body (on a line by itself) to the address ietf-msgtrk-request@imc.org. An archive of the mailing list may be found at <http://www.ietf.org/archive/msgtrk>.

Copyright Notice

Copyright (C) The Internet Society (1999). All Rights Reserved.

Abstract

Customers buying enterprise message systems often ask: Can I track the messages? Message tracking is the ability to find out the path that a particular message has taken through a messaging system and the

current routing status of that message. This document describes the Message Tracking Query Protocol that is used in conjunction with extensions to the ESMTP protocol to provide a complete message tracking solution for the Internet.

NOTE: This is a straw proposal for the Message Tracking Query Protocol.

1. Introduction

The Message Tracking Models and Requirements document [RFC-TRACK-MODEL] discusses the models that message tracking solutions could follow, along with requirements for a message tracking solution that can be used with the Internet-wide message infrastructure. This memo and its companions, [RFC-TRACK-ESMTP] and [RFC-TRACK-TSN], describe a complete message tracking solution that satisfies those requirements. The memo [RFC-TRACK-ESMTP] defines an extension to the SMTP service that provides the information necessary to track messages. This memo defines a protocol that can be used to query the status of messages that have been transmitted on the Internet via SMTP. The memo [RFC-TRACK-TSN] describes the message/tracking-status MIME media type that is used to report tracking status information. Using the model document's terminology, this solution uses active enabling and active requests with both request and chaining referrals.

1.1. 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 [RFC-KEYWORDS].

All syntax descriptions use the ABNF specified by [RFC-ABNF]. Unless otherwise noted, any terminal nodes not defined here are defined in [RFC-ABNF].

2. Basic Operation

The Message Tracking Query Protocol (MTQP) is similar to many other line-oriented Internet protocols, such as [POP3] and [NNTP]. Initially, the server host starts the MTQP service by listening on TCP port ???. When a client wishes to make use of the service, it establishes a TCP connection with the server host. When the connection is established, the MTQP SERVER SENDS A GREETING. The client and MTQP server then exchange commands and responses (respectively) until the connection is closed or aborted.

Commands in MTQP consist of a case-insensitive keyword, possibly followed by one or more parameters. All commands are terminated by a

CRLF pair. Keywords and parameters consist of printable ASCII characters. Keywords and parameters are separated by whitespace (one or more space or tab characters). A command line is limited to 998 characters before the CRLF.

Responses in MTQP consist of a status indicator that indicates success or failure, possibly followed by whitespace and additional information. Successful commands may also be followed by additional lines of data. All response lines are terminated by a CRLF pair and are limited to 998 characters before the CRLF. There are several status indicators: "+OK" indicates success; "+OK+" indicates a success followed by additional lines of data, a multi-line success response; "-TEMP" indicates a temporary failure; "-ERR" indicates a permanent failure; and "-BAD" indicates a protocol error (such as for unrecognized commands).

A multi-line success response may take one of two forms. If the end of the +OK+ line contains the character "{", a number, and the character "}" before the CRLF, then the rest of the response consists of that number of characters.

If the multi-line success response does not end with "{number}", each subsequent line is terminated by a CRLF pair and limited to 998 characters before the CRLF. When all lines of the response have been sent, a final line is sent consisting of a single period (".", decimal code 046) and a CRLF pair. If any line of the multi-line response begins with a period, the line is "dot-stuffed" by prepending the period with a second period. When examining a multi-line response, the client checks to see if the line begins with a period. If so, and other octets other than CRLF follow, the first octet of the line (the period) is stripped away. If so, and if CRLF immediately follows the period, then the response from the MTQP server is ended and the line containing the ".CRLF" is not considered part of the multi-line response.

An MTQP server MUST respond to an unrecognized, unimplemented, or syntactically invalid command by responding with a negative -BAD status indicator. A server MUST respond to a command issued when the session is in an incorrect state by responding with a negative -ERR status indicator.

An MTQP server MAY have an inactivity autologout timer. Such a timer MUST be of at least 10 minutes' duration. The receipt of any command from the client during that interval should suffice to reset the autologout timer.

3. Initialization

Once the TCP connection has been opened by an MTQP client, the MTQP server issues an initial status response indicates its readiness. If

the status response is positive (+OK or +OK+), the client may proceed with other commands.

If the server has any options enabled, they are listed as the multi-line response of the initial status response, one per line. An option specification consists of an identifier, optionally followed by option-specific parameters. An option specification may be continued onto additional lines by starting the continuation lines with white space.

No options are defined in this document.

Example #1 (no options):

S: +OK MTQ server ready

Example #2 (service temporarily unavailable):

S: -TEMP Service down for admin, call back later

Example #3 (service permanently unavailable):

S: -ERR Service down

Example #4 (alternative for no options):

S: +OK+ MTQ server ready

S: .

Example #5 (options available):

S: +OK+ MTQ server ready

S: Option1 parameters

S: Option2

S: Option3 a very long

S: list of parameters

S: .

4. TRACK Command

Syntax:

"TRACK" 1*WS tracking-id 1*WS authorization-cookie *WS CRLF

tracking-id = TBD

authorization-cookie = TBD

When the client issues the TRACK command, the MTQP server retrieves tracking information about an email message. A successful response MUST be multi-line, consisting of a [[MIME](#)] mail message whose default content-type is message/tracking-status, as defined in [[RFC-TRACK-TSN](#)]. This message contains the tracking information about the email message that used the given tracking-id. The tracking-id and authorization-

cookie are defined in [[RFC-TRACK-ESMTP](#)]. The authorization-cookie is expressed in hexadecimal.

Example #6

```
C: TRACK <tracking-id> 1234567890ABCDEF
S: +OK+ Tracking information follows
S: Content-Type: message/tracking-status
S:
S: ... details go here ...
S: .
```

5. NOOP Command

Syntax:

```
"NOOP" opt-text CRLF
```

When the client issues the NOOP command, the MTQP server resets the inactivity autologout timer. The server **MUST** respond with a successful response (+OK or +OK+). All parameters to the NOOP command are ignored.

6. QUIT Command

Syntax:

```
"QUIT" *WS CRLF
```

When the client issues the QUIT command, the MTQP session terminates. The QUIT command has no parameters. The server **MUST** respond with a successful response. The client may close the session from its end immediately after issuing this command.

7. Pipelining

The MTQP client may elect to transmit groups of MTQP commands in batches without waiting for a response to each individual command. The MTQP server **MUST** process the commands in the order received. The following two examples are identical:

Example #7

```
C: TRACK <tracking-id> 1234567890ABCDEF
S: +OK+ Tracking information follows
S:
S: ... details go here ...
S: .
C: NOOP
S: +OK Status okay
```

Example #8

```
C: TRACK <tracking-id> 1234567890ABCDEF
```



```
C: NOOP
S: +OK+ Tracking information follows
S:
S: ... details go here ...
S: .
S: +OK Status okay
```

8. URL Format

The MTQP URL scheme is used to designate MTQP servers on Internet hosts accessible using the MTQP protocol. An MTQP URL takes one of the following forms:

```
mtqp://<mserver>/track/<tracking-id>:<cookie>
mtqp://<mserver>:<port>/track/<tracking-id>:<cookie>
```

The first form is used to refer to an MTQP server on the standard port, while the second form specifies a non-standard port. Both of these forms specify that the TRACK command is to be issued using the given tracking id and authorization cookie. The path element "/track/" is case insensitive, but the tracking id may not be.

8.1. MTQP URL Syntax

This is an ABNF description of the MTQP URL. Terminal nodes not defined here are defined in either [[RFC-URL](#)] or [[RFC-ABNF](#)].

```
mtqp-url = "mtqp://" net_loc "/track/" tracking-id ":" cookie

tracking-id = TBD

cookie = 16HEXDIG
```

9. IANA Considerations

TBD - registering extensions

10. Security Considerations

Security considerations discussed in [[RFC-TRACK-MODEL](#)] and [RFC-TRACK-ESMTP] are relevant.

11. Protocol Syntax

This is an ABNF description of MTQP.

```
command-response = success-response / temp-response / error-response /
bad-response
```

```
temp-response = "-TEMP" opt-text CRLF
```


opt-text = [WSP *(VCHAR / WSP)]

error-response = "-ERR" opt-text CRLF

bad-response = "-BAD" opt-text CRLF

success-response = single-line-success / multi-line-success / multi-char-success

single-line-success = "+OK" opt-text CRLF

multi-char-success = "+OK+" opt-text "{" 1*DIGIT "}" *WSP CRLF *OCTET
; the number of characters specified by {number} ; must be
sent

multi-line-success = "+OK+" opt-text CRLF *dataline dotcrlf

dataline = *998OCTET CRLF

dotcrlf = "." CRLF

option-list = *option-line

option-line = rulename opt-text *[CRLF WSP opt-text] CRLF

12. References

[MIME] [RFC 2045](#), N. Freed & N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", November 1996.

[RFC-821] STD 10, [RFC 821](#), J. Postel, "Simple Mail Transfer Protocol", University of Southern California / Information Sciences Institute, August 1982.

[RFC-822] STD 11, [RFC 822](#), D. Crocker, "Standard for the Format of ARPA Internet Text Messages", University of Delaware, August 1982.

[RFC-ABNF] [RFC 2234](#), D. Crocker, Editor, and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", November 1997.

[RFC-ESMTP] [RFC 1651](#), J. Klensin, N. Freed, M. Rose, E. Stefferud, and D. Crocker, "SMTP Service Extensions", Silicon Graphics, Inc., July 1994.

[RFC-KEYWORDS] [RFC 2119](#), S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC-MD5] [RFC 1321](#), R. Rivest, "The MD5 Message-Digest Algorithm", April 1992.

[RFC-TRACK-MODEL] [draft-ietf-msgtrk-model-02.txt](#), T. Hansen, K. Lin, "Message Tracking Models and Requirements", AT&T Laboratories, Lotus Development Corporation, ??? 2000.

[RFC-TRACK-ESMTP] [draft-ietf-msgtrk-smtpext](#)-.txt, E. Allman, "SMTP Service Extension for Message Tracking", Sendmail, Inc., ??? 2000.

[RFC-TRACK-TSN] [draft-ietf-msgtrk-trkstat-00.txt](#), E. Allman, "The Message/Tracking-Status MIME Extension", Sendmail, Inc., ??? 2000.

[RFC-URL] [RFC 1808](#), R. Fielding, "Relative Uniform Resource Locators", June 1995.

13. Authors' Addresses

Tony Hansen
AT&T Laboratories
Lincroft, NJ 07738
USA

Phone: +1.732.576.3207
E-Mail: tony@att.com

14. Full Copyright Statement

Copyright (C) The Internet Society (1999). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organisations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING

TASK FORCE DISCLAIMS 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.

This document expires January 1, 2001.