Network Working Group   T. Hansen
Request for Comments: 3888   AT&T Laboratories
Category: Informational   September 2004

Message Tracking Model and Requirements

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

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 provides a model of message tracking that can be used for understanding the Internet-wide message infrastructure and to further enhance those capabilities to include message tracking, as well as requirements for proposed message tracking solutions.

1. Problem Statement

Consider sending a package through a package delivery company. Once you've sent a package, you would like to be able to find out if the package has been delivered or not, and if not, where that package currently is and what its status is. Note that the status of a package may not include whether it was delivered to its addressee, but just the destination. Many package carriers provide such services today, often via a web interface.

Message tracking extends that capability to the Internet-wide message infrastructure, analogous to the service provided by package carriers: the ability to quickly locate where a message (package) is, and to determine whether or not the message (package) has been delivered to its final destination. An Internet-standard approach will allow the development of message tracking applications that can operate in a multi-vendor messaging environment, and will encourage the operation of the function across administrative boundaries.

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 BCP 14, RFC 2119 [RFC-KEYWORDS].

2. Definitions

The following terms are relevant to message tracking. The terms Tracking User Agent and Tracking Server are new, while all other terms have been collected here from other sources.

Originating Mail User Agent (MUA)
: The originating mail user agent is the software used to compose and originate a message. It is the software sitting on a person's desktop.

Originating Mail Submission Agent (MSA)
: The Mail Submission Agent accepts a message from a User Agent, adds or modifies it as required for Internet standards and/or site policy, and injects the message into the network. The MSA may be the initial MTA or may hand off the message to an MTA.

Message Transfer Agent (MTA)
: A Message Transfer Agent accepts a message and moves it forward towards its destination. That destination may be local or reached via another MTA. It may use a local queue to store the message before transferring it further. Any MTA may generate a Non-Delivery Notification.

Intermediate Message Transfer Agent (MTA)
: An Intermediate MTA is an MTA that accepts a message for transfer somewhere else.

Final Message Transfer Agent (MTA)
: A Final MTA is an MTA that accepts a message for local delivery. It is the final place that a message is accepted. The final MTA is what sends any Delivery Status Notifications (DSNs). (Intermediate MTA's may also send a DSN if it relays to a non-DSN aware MTA.)

Foreign Message Transfer Agent
: A foreign MTA provides delivery of messages using other protocols than those specified for Internet mail, such as an X.400 mail system.

Gateway Message Transfer Agent (GW-MTA)
A gateway MTA accepts a message for transfer to a foreign MTA outside of the Internet protocol space.

Local Delivery Agent (LDA)
The local Delivery Agent delivers the message to the local message store. (The MTA and LDA are often combined into the same program.)

Delivery Status Notification (DSN)
A Delivery Status Notification [RFC-DSN] is produced by an MTA when a message is unsuccessfully delivered, either to its next hop or the final message store, or when it is successfully delivered, either to a foreign MTA, to a local delivery agent, or a non-DSN aware MTA. Positive notifications are only performed [RFC-ESMTP-DSN] when specifically requested.

Non-Delivery Notification (NDN)
A non-delivery notification is a special form of DSN indicating unsuccessful delivery.

Message Disposition Notification (MDN)
A Message Disposition Notification is used to report the disposition of a message after it has been successfully delivered to a recipient.

Tracking User Agent (TUA)
A tracking user agent wants to find information on a message on the behalf of a user. It is the requestor or initiator of such a request. (The MUA and TUA could be combined into the same program.)

Tracking Server
A tracking server provides tracking information to a tracking client. It is the repository of the information about a message for the traversal through a particular MTA. (The tracking server and MTA may run on the same system.)

3. Entities

The entities involved in message tracking are: message user agents, message submission agents, message transfer agents, tracking user agents and tracking servers.

4. Requirements

These are requirements that any message tracking solution must be able to satisfy:

The message tracking solution:

** MUST scale to the internet.

** MUST be easy to deploy.

** SHOULD maximize the reuse of existing, already deployed technology and infrastructure.

** If possible, SHOULD extend existing protocols and not invent new ones.

** SHOULD have a low implementation cost. (This makes it easy to incorporate into existing products.)

** MUST restrict tracking of a message to the originator of the message (or a delegate).

** MUST be able to do authentication.

** MAY allow an originator to delegate this responsibility to a third party.

** SHOULD have the property that they would allow per-message delegation of the tracking responsibility.

** MUST require a tracking user agent to prove that they are permitted to request the tracking information.

** MUST be able to uniquely identify messages.

** MUST require every message to have unique identification.

5. Interaction Models

There are several models by which tracking of messages can be enabled, by which messages can be tracked, and by which information can be requested and gathered.

5.1. Tracking Enabling Models

Either the envelope or message header must contain enough information to track a message and securely retrieve information about the message. Any message that does not have enough information to track it is by definition not trackable.

If there is not enough information available in current standard envelopes or message headers, then the current standards will need to be extended. Either the MUA or MSA must determine the additional information and enable the tracking by adding the additional information to either the envelope or header.

This leads to two tracking enabling models: passive enabling and active enabling.

5.1.1. Passive Enabling Model

The "passive enabling" model assumes that there is sufficient information available. No UA or MSA interaction occurs to turn tracking on; it is on by default.

5.1.2. Active Enabling Model

The "active enabling" model requires that the MUA and MSA exchange information when the message is submitted. This exchange indicates that logging of the message's traversal should be performed, as well as providing enough additional information to allow the message to be tracked. This information will need to be passed on to subsequent MTAs as needed.

5.2. Tracking Request Models

There are several models by which tracking information may be requested.

5.2.1. Passive Request Model

The "passive request" model requires active enabling to indicate that some form of tracking is to be performed. The tracking information can be sent back immediately (as a form of telemetry) or sent to a 3rd party for later retrieval.

5.2.2. Passive Request Tracking Information

Forms of passive tracking information that could potentially be requested are as follows. Note that mechanisms already exist for requesting the information marked with a (+). The references for such mechanisms are listed at the end of each such entry.

** send a DSN of a message arriving at an intermediate MTA

** (+) send a DSN of a message being rejected while at an intermediate MTA [RFC-DSN]

** (+) send a DSN of a message leaving an intermediate MTA and going to another MTA [RFC-DELIVER-BY]

** send a DSN of a message arriving at a final MTA

** (+) send a DSN of a message being rejected while at a final MTA [RFC-DSN]

** (+) send a DSN of a message being delivered to a user's message store [RFC-DSN]

** (+) send a DSN of a message being delivered to a foreign MTA [RFC-DSN]

** (+) send an MDN of a message being read by an end user [RFC-MDN]

5.3. Active Request Model

The "active request" model requires an active query by a user's user agent to the MSA, intermediate MTAs and final MTA, or to a third party, to find the message's status as known by that MTA. Active request will work with either passive enabling or active enabling.

5.3.1. Server Chaining vs. Server Referrals

When a tracking server has been asked for tracking information, and the message has been passed on to another MTA of which this tracking server has no tracking knowledge, there are two modelling choices:

** the first tracking server will contact the next tracking server to query for status and pass back the combined status (server chaining), or

** the first tracking server will return the address of the next MTA and the tracking client has the responsibility of contacting the next tracking server (server referrals).

5.3.2. Active Request Tracking Information

Forms of active tracking information that could potentially be requested are as follows. (Note that no mechanisms currently exist for requesting such information.)

** the message has been queued for later delivery

** the message was delivered locally

** the message was delivered to another MTA,

** the message was delivered to a foreign MTA

** ask a different tracking server,

** I know but can't tell you,

** I don't know.

5.4. Combining DSN and MDN Information with Message Tracking Information

The information that would be retrieved by message tracking and the information that is returned for DSN and MDN requests all attempt to answer the question of "what happened to message XX"? The information provided by each is complimentary in nature, but similar. A tracking user agent could use all three possible information sources to present a total view of the status of a message.

Both DSN and MDN notifications utilize the formats defined by RFC 3462 [RFC-REPORT]. This suggests that the information returned by message tracking solutions should also be similar.

6. Security Considerations

6.1. Security Considerations Summary

Security vulnerabilities are detailed in [RFC-MTRK-ESMTP], [RFC-MTRK-TSN] and [RFC-MTRK-MTQP]. These considerations include:

** vulnerability to snooping or replay attacks when using unencrypted sessions

** a dependency on the randomness of the per-message secret

** reliance on TLS

** man-in-the-middle attacks

** reliance on the server maintaining the security level when it performs chaining

** denial of service

** confidentiality concerns

** forgery by malicious servers

6.2. Message Identification and Authentication

This is a security model for message identification and authentication that could be deployed. (There may be others.)

A Tracking User Agent must prove that they are permitted to request tracking information about a message. Every [RFC-822]-compliant message is supposed to contain a Message-Id header. One possible mechanism is for the originator to calculate a one-way hash A from the message ID + time stamp + a per-user secret. The user then calculates another one-way hash B to be the hash of A. The user includes B in the submitted message, and retains A. Later, when the user makes a message tracking request to the messaging system or tracking entity, it submits A in the tracking request. The entity receiving the tracking request then uses A to calculate B, since it was already provided B, verifying that the requestor is authentic. In summary,

    A = H(message ID + time stamp + secret)

    B = H(A)

Another possible mechanism for A is to ignore the message ID and time stamp and just use a one-way hash from a large (>128 bits) random number. B would be calculated as before. In summary,

    A = H(large-random-number)

    B = H(A)

This is similar in technique to the methods used for One-Time Passwords [RFC-OTP]. The success of these techniques is dependent on the randomness of the per-user secret or the large random number, which can be incredibly difficult in some environments.

If the originator of a message were to delegate his or her tracking request to a third party by sending it A, this would be vulnerable to snooping over unencrypted sessions. The user can decide on a message-by-message basis if this risk is acceptable.

7. Informational References

[RFC-822] Crocker, D., "Standard for the format of ARPA Internet text messages", STD 11, RFC 822, August 1982.

[RFC-DELIVER-BY] Newman, D., "Deliver By SMTP Service Extension", RFC 2852, June 2000.

[RFC-DSN] Moore, K., and G. Vaudreuil, "An Extensible Message Format for Delivery Status Notifications", RFC 3464, January 2003.

[RFC-ESMTP-DSN] Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, January 2003.

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

[RFC-MDN] Hansen, T. and G. Vaudreuil, Eds., "Message Disposition Notifications", RFC 3798, May 2004.

[RFC-OTP] Haller, N., Metz, C., Nesser, P. and M. Straw, "A One-Time Password System", STD 61, RFC 2289, February 1998.

[RFC-REPORT] Vaudreuil, G., "The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages", RFC 3462, January 2003.

[RFC-MTRK-ESMTP] Allman, E. and T. Hansen, "SMTP Service Extension for Message Tracking", RFC 3885, September 2004.

[RFC-MTRK-TSN] Allman, E., "The Message/Tracking-Status MIME Extension", RFC 3886, September 2004.

[RFC-MTRK-MTQP] Hansen, T., "Message Tracking Query Protocol", RFC 3887, September 2004.

8. Acknowledgements

This document is the product of input from many people and many sources, including all of the members of the Message Tracking Working Group: Philip Hazel, Alexey Melnikov, Lyndon Nerenberg, Chris Newman, and Gregory Neil Shapiro. It owes much to earlier work by Gordon Jones, Bruce Ernst, and Greg Vaudreuil. In particular, I'd like to also thank Ken Lin for his considerable contributions to the early versions of the document.

9. Author's Address

Tony Hansen
AT&T Laboratories
Middletown, NJ 07748
USA

Phone: +1.732.420.8934
EMail: tony+msgtrk@maillennium.att.com

10. Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.