

MARID / SMTP  
Internet-Draft  
Expires: November 7, 2004

D. Crocker  
Brandenburg InternetWorking  
May 09, 2004

Internet Mail Architecture  
draft-crocker-email-arch-00

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/lid-abstracts.txt>.

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

This Internet-Draft will expire on November 7, 2004.

Copyright Notice

Copyright (C) The Internet Society (2004). All Rights Reserved.

Abstract

Over its thirty year history, Internet mail has undergone significant changes in scale and complexity. The first standardized architecture for email specified a simple split between the user world and the transmission world, in the form of Mail User Agents (MUA) and Mail Transfer Agents (MTA). Over time each of these has divided into multiple, specialized modules. Public discussion and agreement about the nature of the changes to Internet mail has not kept pace, and abuses of the Internet mail service have brought these issues into stark relief. This draft offers clarifications and enhancements, to provide a more consistent base for community discussion of email service problems and proposed email service enhancements.

Internet-Draft

MailArch

May 2004

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">1.1</a>	Service Overview . . . . .	<a href="#">3</a>
<a href="#">1.2</a>	Discussion venue . . . . .	<a href="#">4</a>
<a href="#">2.</a>	Email Identities . . . . .	<a href="#">5</a>
<a href="#">2.1</a>	Mailbox Addresses . . . . .	<a href="#">5</a>
<a href="#">2.1.1</a>	Global Standards for Local-Part . . . . .	<a href="#">5</a>
<a href="#">2.1.2</a>	Scope of Email Address Use . . . . .	<a href="#">5</a>
<a href="#">2.2</a>	Domain Names . . . . .	<a href="#">6</a>
<a href="#">2.3</a>	Identity Reference Convention . . . . .	<a href="#">6</a>
<a href="#">3.</a>	Email System Architecture . . . . .	<a href="#">6</a>
<a href="#">3.1</a>	Architectural Components . . . . .	<a href="#">7</a>
<a href="#">3.1.1</a>	Mail User Agent (MUA) . . . . .	<a href="#">7</a>
<a href="#">3.1.2</a>	Mail Submission Agent (MSA) . . . . .	<a href="#">9</a>
<a href="#">3.1.3</a>	Mail Transfer Agent (MTA) . . . . .	<a href="#">11</a>
<a href="#">3.1.4</a>	Mail Delivery Agent (MDA) . . . . .	<a href="#">11</a>
<a href="#">3.2</a>	Operational Configuration . . . . .	<a href="#">12</a>
<a href="#">3.3</a>	Layers of Identity References . . . . .	<a href="#">13</a>
<a href="#">4.</a>	Message Data . . . . .	<a href="#">13</a>
<a href="#">4.1</a>	Envelope . . . . .	<a href="#">13</a>
<a href="#">4.2</a>	Message Headers . . . . .	<a href="#">13</a>
<a href="#">4.3</a>	Body . . . . .	<a href="#">14</a>
<a href="#">5.</a>	Two Levels of Store-And-Forward . . . . .	<a href="#">14</a>
<a href="#">5.1</a>	MTA Relaying . . . . .	<a href="#">14</a>
<a href="#">5.2</a>	MUA Forwarding . . . . .	<a href="#">14</a>
<a href="#">5.2.1</a>	MUA Basic Forwarding . . . . .	<a href="#">14</a>
<a href="#">5.2.2</a>	MUA Re-Sending . . . . .	<a href="#">15</a>
<a href="#">5.2.3</a>	MUA Reply . . . . .	<a href="#">16</a>
<a href="#">5.2.4</a>	MUA Gateways . . . . .	<a href="#">17</a>
<a href="#">5.2.5</a>	MUA Alias Handling . . . . .	<a href="#">18</a>
<a href="#">5.2.6</a>	MUA Mailing Lists . . . . .	<a href="#">19</a>
<a href="#">6.</a>	Security Considerations . . . . .	<a href="#">21</a>
<a href="#">7.</a>	References . . . . .	<a href="#">21</a>
	Author's Address . . . . .	<a href="#">23</a>
<a href="#">A.</a>	Acknowledgements . . . . .	<a href="#">23</a>
	Intellectual Property and Copyright Statements . . . . .	<a href="#">24</a>

---

Internet-Draft

MailArch

May 2004

## [1.](#) Introduction

Over its thirty year history, Internet mail has undergone significant changes in scale and complexity. The first standardized architecture for email specified a simple split between the user world and the transmission world, in the form of Mail User Agents (MUA) and Mail Transfer Agents (MTA). Over time each of these has sub-divided into more specialized modules.

The basic style and use of names, addresses and message structure have remained remarkably constant. However each has benefited from significant elaborations. Public discussion and agreement about the nature of these changes has not kept pace, and abuses of the Internet mail service have brought these issues into stark relief.

The current draft seeks to:

1. document changes that have taken place in refining the email model
2. clarify functional roles for the architectural components
3. clarify identity-related issues, across the email service
4. provide a common venue for further defining and citing modern Internet mail architecture

### [1.1](#) Service Overview

End-to-end Internet mail exchange is accomplished by using a standardized infrastructure comprising:

1. an email object

2. global addressing
3. a connected sequence of point-to-point transfer mechanisms
4. no prior arrangement between originator and recipient
5. no prior arrangement between point-to-point transfer services, over the open Internet

The end-to-end portion of the service is the message. Broadly the message, itself, is divided between handling control information and user message payload.

A precept to the design of Internet mail is to permit interoperability with no prior, direct administrative arrangement between the participants. That is, all participants rely on having the core services be universally supported, either directly or through gateways that translate between Internet mail standards and other email conventions.

For localized environments (edge networks) prior, administrative arrangement can include access control, routing and lookup service configuration. One change to local environments, occurring in recent years, is an increased requirement for authentication or, at least, accountability. In these cases, the server performs explicit validation of the client's identity.

## 1.2 Discussion venue

NOTE: This document is the work of a single person, about a topic with considerable diversity of views. It is certain to be incomplete and inaccurate. Some errors simply need to be reported; they will get fixed. Others need to be discussed by the community, because the real requirement is to develop common community views. To this end, please treat the draft as a touchstone for public discussion.

Discussion about this document should be directed to the:

<<mailto:ietf-smtp@imc.org>>

mailing list. Located at <<http://www.imc.org/ietf-smtp/index.html>> the ietf-smtp mailing list is the most active, long-standing venue for discussing email architecture. Although this list is primarily for discussing only the SMTP protocol, it is recommended that discussion of this draft take place on that mailing list. This list tends to attend to end-to-end infrastructure and architecture issues more than other email-related mailing lists.

- o The <mailto:ietf-822@imc.org> list also is pertinent <<http://www.imc.org/ietf-smtp/index.html>>. However it's focus is on the message, itself, so that transfer issues are typically excluded. In addition, this list has not been very active recently.
- o A currently active mailing list, likely to impact Internet mail architecture, is <mailto:ietf-mxcomp@imc.org>. This list is devoted to matters of spam control, so that underlying matters of Internet mail architecture are probably best deferred to a more general list, such as ietf-smtp.
- o Also currently active is the <mailto:lemonade@ietf.org>, which is

Crocker

Expires November 7, 2004

[Page 4]

---

Internet-Draft

MailArch

May 2004

considering enhancements for interaction between thin MUAs and MSAs.

## 2. Email Identities

Internet mail uses two forms of identity. The most common is the mailbox address <addr-spec> [[RFC2822](#)]. The other form is the <domain name> [[RFC1034](#)].

### 2.1 Mailbox Addresses

An addr-spec has two distinct parts, divided by an at-sign ("@"). The right-hand side contains a globally interpreted name for an administrative domain. This domain name might refer to an entire organization, or to a collection of machines integrated into a homogeneous service, or to a single machine. Domain names are defined and operated through the DNS [[RFC1034](#)], [[RFC1035](#)].

The left-hand side of the at-sign contains a string that is globally opaque and is called the <local-part>. It is to be interpreted only

by the entity specified in the address's right-hand side. All other entities must treat the local-part as a uninterpreted, literal string and must preserve all of its original details. As such, its distribution is equivalent to sending a "cookie" that is only interpreted upon being returned to its originator.

#### [2.1.1](#) Global Standards for Local-Part

A small class of addresses have an elaboration on basic email addressing, with a standardized, global schema for the local-part. These are conventions between originating end-systems and recipient gateways, and they are invisible to the public email transfer infrastructure. When an originator is explicitly sending via a gateway out of the Internet, there are coding conventions for the local-part, so that the originator can formulate instructions for the gateway. Standardized examples of this are the telephone numbering formats for VPIM [[RFC2421](#)], such as "+16137637582@vpim.example.com", and iFax [[RFC2304](#)], such as "FAX=+12027653000/T33S=1387@ifax.example.com".

#### [2.1.2](#) Scope of Email Address Use

Email addresses are being used far beyond their original email transfer and delivery role. In practical terms, email strings have become a common form of user identity on the Internet. What is essential, then, is to be clear about the nature and role of an identity string in a particular context and to be clear about the

entity responsible for setting that string.

## [2.2](#) Domain Names

A domain name is a global reference to an Internet resource, such as a host, a service or a network. A name usually maps to an IP Address. A domain name can be administered to refer to individual users, but this is not common practice. The name is structure as a hierarchical sequence of sub-names, separated by dots (".").

When not part of a mailbox address, a domain name is used in Internet mail to refer to a node that took action upon the message, such as providing the administrative scope for a message identifier, or performing transfer processing.

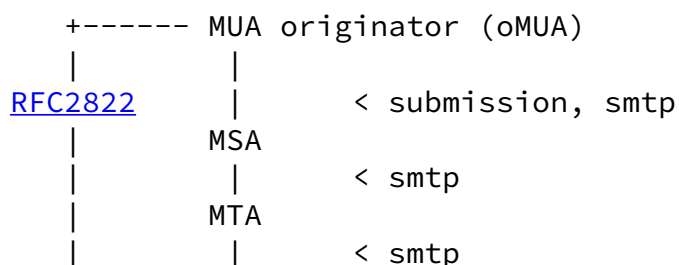
### 2.3 Identity Reference Convention

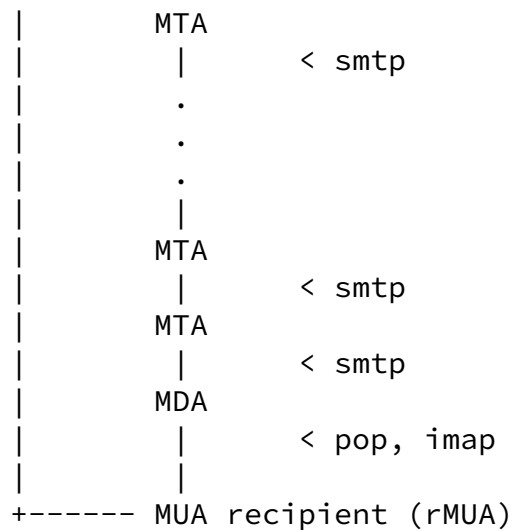
In this document, fields containing identity references are labeled in a two-part, dotted notation, that cites document defining them and the field containing them. Hence, <[RFC2822](#).From> is the From field in an email content header, and <[RFC2821](#).MailFrom> is the address in the SMTP "Mail From" command.

### 3. Email System Architecture

NOTE: A discussion about any interesting system architecture is often complicated by confusion between architecture versus implementation. An architecture defines the conceptual functions of a service, divided into discrete conceptual modules. An implementation of that architecture may combine or separate architectural components, as needed for a particular operational environment. It is important not to confuse the engineering decisions that are made to implement a product, with the architectural abstractions used to define conceptual functions.

Modern Internet email architecture distinguishes four types of components, arranged to support a store-and-forward service architecture:





Software implementations of these architectural components often compress them, such as having the same software do MSA, MTA and MDA functions. However the requirements for each of these components of the architecture are becoming more extensive. So, their separation is increasingly common.

## 3.1 Architectural Components

### 3.1.1 Mail User Agent (MUA)

An <MUA> works on behalf of end-users and end-user applications. It is their "representative" within the email service.

At the origination side of the service, the <oMUA> is used to create a message and perform initial "submission" into the transfer infrastructure, via an <MSA>. It may also perform any creation- and posting-time archival. An MUA outbox is part of the origination-side MUA.

The inbox and other folders are part of the recipient-side <rMUA> that works on behalf of the end-user to process received mail. This includes generating user-level return control messages, display and disposition of the received message, and closing or expanding the user communication loop, by initiating replies and forwarding new messages.



An MUA may, itself, have a distributed architecture, such as implementing a "thin" user interface module on a limited end-user device, with the bulk of the MUA functionality operated remotely on a more capable server. An example of such an architecture might use IMAP [[RFC3501](#)] for most of the interactions between an MUA client and an MUA server.

A special class of MUA functions perform message forwarding, as discussed in the MUA Forwarding ([Section 5.2](#)) section.

Identity fields set by the MUA include:

Identity	Role
<a href="#">RFC2822</a> .From	Names and addresses for author(s) of the message content are listed in the From header
<a href="#">RFC2822</a> .Reply-To	If a message recipient sends a message that would otherwise use the <a href="#">RFC2822</a> .From field information in the original message, they are to use the contents of the <a href="#">RFC2822</a> .Reply-To field instead. In other words, this field is a direct override of the From field, for responses from recipients.
<a href="#">RFC2822</a> .Sender	This specifies the address responsible for submission into the transfer service. For efficiency, this field should be omitted if it contains the same address as <a href="#">RFC2822</a> .From. However this does not mean there is no Sender specified. Rather, it means that that header is virtual and that the address in the From field must be used. Specification of the error return addresses (the "bounces" address, contained in <a href="#">RFC2821</a> .MailFrom) is made by the Sender. Typically the bounce address is the same as the Sender address. However

Internet-Draft

MailArch

May 2004

	some usage scenarios require it to be different.
<a href="#">RFC2822</a> .To, <a href="#">RFC2822</a> .CC	These specify MUA recipient addresses. The distinction between To and CC is subjective. Generally, a To addressee is considered primary and is expected to take action on the message. A CC addressee typically receives a copy only for their information.
<a href="#">RFC2822</a> .BCC	A message might be copied to an addressee who is not to be disclosed to the <a href="#">RFC2822</a> .TO or <a href="#">RFC2822</a> .CC recipients. The BCC header indicates a message copy to such a recipient. Typically, the field lists no addresses or only lists the address of the single recipient receiving the copy. This ensures that even other BCC recipients do not know of each other.

Table 1: Message Identities

### [3.1.2](#) Mail Submission Agent (MSA)

An <MSA> accepts the message submission from the oMUA and conditions it for insertion into the global email transfer network, according to the policies of the hosting network and the requirements of Internet standards. It implements a server function to MUAs and a client function to MTAs (or MDAs).

Examples of MSA-styled functions, in the world of paper mail, might range across the very different capabilities of administrative assistants, postal drop boxes, and post office front-counter employees.

The MUA/MSA interface can be implemented within single host and use private conventions for their interactions. Historically,

standards-based MUA/MSA interactions have used SMTP [[RFC2821](#)]. However a recent alternative is SUBMISSION [[RFC2476](#)]. Although SUBMISSION derives from SMTP, it operates on a separate TCP port, and will typically impose distinct requirements, such as access authorization.

Identities set by the MSA include:

Identity	Role
<a href="#">RFC2821</a> .HELO or <a href="#">RFC2821</a> .EHLO	The MSA may specify its hosting domain identity for the SMTP HELO or EHLO command operation.
<a href="#">RFC2821</a> .MailFrom	This is an end-to-end string that specifies an email address for receiving return control information, such as "bounces". The name of this field is misleading, because it is not required to specify either the author or the agent responsible for submitting the message. Rather, the agent responsible for submission specifies the <a href="#">RFC2821</a> .MailFrom address. Ultimately the simple basis for deciding what address needs to be in the <a href="#">RFC2821</a> .MailFrom is to determine what address needs to be informed about transmission-level problems (and, possibly, successes.)
<a href="#">RFC2821</a> .Rcpt-To	This specifies the MUA inbox address of a recipient. The string might not be visible in the message content headers. For example, the message destination address headers, such as <a href="#">RFC2822</a> .To, might specify a mailing list address, while the <a href="#">RFC2821</a> .Rcpt-To

	address specifies a member of that list.
<a href="#">RFC2821</a> .Received	An MSA may record a Received header, to indicate initial submission trace information, including originating host and MSA host domain names and/or IP Addresses.

Table 2: MSA Identities

### [3.1.3](#) Mail Transfer Agent (MTA)

An <MTA> relays a message to another other MTA or to an <MDA>, in a point-to-point exchange. Relaying is performed by a sequence of MTAs, until the message reaches its destination MDA. Hence an MTA implements both client and server MTA functionality.

The basic store-and-forward functionality of an MTA is similar to that of a packet switch or router. However email objects are typically much larger than the payload of a packet or datagram, and the end-to-end latencies are typically much higher. Contrary to typical packet switches (and Instant Messaging services) Internet mail MTAs typically store messages in a manner that allows recovery across services interruptions, such as host system shutdown.

Internet mail primarily uses SMTP [[RFC2821](#)], [[RFC0821](#)] to effect point-to-point transfers between peer MTAs. Other transfer mechanisms include Batch SMTP [[RFC2442](#)] and ODMR [[RFC2645](#)]

An important characteristic of MTA-MTA communications, over the open Internet, is that they do not require prior arrangement between the independent administrations operating the different MTAs. Given the importance of spontaneity and serendipity in the world of human communications, this lack of prearrangement, between the participants, is a core benefit of Internet mail and remains a core requirement for it.

### [3.1.4](#) Mail Delivery Agent (MDA)

The <MDA> delivers email to the recipient's inbox.

An MDA can provide distinctive, address-based functionality, made possible by its detailed knowledge of the properties of the destination address. This knowledge might also be present earlier in an MTA relaying sequence that ends with the MDA, such as at an organizational gateway. However it is required for the MDA, if only because the MDA must know where to store the message. This knowledge is used to achieve differential handling of messages.

Using Internet protocols, delivery is effected with POP [[RFC1939](#)], IMAP [[RFC3501](#)]. SMTP permits "push" delivery to the recipient system, at the imitative of the upstream email service. POP is used for "pull" delivery at the initiative of the recipient system. Notably, SMTP and POP effect a transfer of message control from the email service to the recipient host. In contrast, IMAP provides on-going, interactive access to a message store, and does not effect a transfer of message control to the end-user host. Instead, control stays with the message store host that is being access by the user.

Identities set by the MSA include:

Identity	Role
<a href="#">RFC2821</a> .HELO or <a href="#">RFC2821</a> .EHLO	The MTA may specify its hosting domain identity for the SMTP HELO or EHLO command operation.
<a href="#">RFC2822</a> .Received	An MTA must record a Received header, to indicate trace information, including source host and receiving host domain names and/or IP Addresses.

Table 3: MTA Identities

## [3.2](#) Operational Configuration

Mail service components can be arranged into numerous organizational structures, each with independent software and administration. One

common arrangement is to distinguish:

1. an open, core, global email transfer infrastructure
2. independent transfer services in networks at the edge of the core
3. end-user services

Edge networks may use proprietary email standards. However the distinction between "public" network and edge network transfer services is primarily significant because it highlights the need for concern over interaction and protection between independent administrations. In particular, this distinction calls for additional care in assessing transitions of responsibility, as well as the accountability and authorization relationships among participants in email transfer.

On the other hand, real-world operations of Internet mail environments do impose boundaries such as access control at organizational firewalls to the Internet. It should be noted that the current Internet Mail architecture offers no special constructs for these configuration choices. The current design of Internet mail is for a seamless, end-to-end store-and-forward sequence. It is possible that the architectural enhancement will not require new protocols, but rather will require clarification of best practises, as exemplified by a recent effort [[I-D.hutzler-spamops](#)]

### [3.3](#) Layers of Identity References

For a message in transit, the core identity fields combine into

Layer	Field	Set By
Message Content	MIME Headers	Author
Message Headers	From	Author
	Sender	Agent of Author
	Reply-To	Author
	To, CC, BCC	Author
	Received	MSA, MTA, MDA
	Return-Path	MDA from MailFrom
SMTP	HELO	Latest MTA Client

	MailFrom	Agent of Author	
	RCPT-TO	MSA	
IP	IP Address	Latest MTA Client	
+-----+	+-----+	+-----+	+-----+

## [4. Message Data](#)

### [4.1 Envelope](#)

Information that is directly used or produced by the email transfer service is called the "envelope". It controls and records handling activities by the transfer service. Internet mail has a fragmented framework for handling this "handling" information. The envelope exists partly in the transfer protocol SMTP [[RFC2821](#)] and partly in the message object [[RFC2822](#)].

Direct envelope addressing information, and optional transfer directives, are carried in-band by MTAs. All other envelope information, such as trace records, is carried within the content headers. Upon delivery, SMTP-level envelope information is typically encoded within additional content headers, such as Return-Path and Received (From and For).

### [4.2 Message Headers](#)

Headers are attribute/value pairs covering an extensible range of email service, user content and user transaction meta-information. The core set of headers is defined in [[RFC2822](#)], [[RFC0822](#)]. It is common to extend this set, for different applications. A complete set of registered headers is being developed through [[ID-HDR-Reg](#)].

One danger with placing additional information in headers is that

gateways often alter or delete them.

### [4.3 Body](#)

The body of a message might simply be lines of ASCII text or it might be structured into a composition of multi-media, body-part attachments, using MIME [[RFC2045](#)], [[RFC2046](#)], [[RFC2047](#)], [[RFC2048](#)], and [[RFC2049](#)]. It should be noted that MIME structures each

body-part into a recursive set of MIME Header meta-data and MIME Content sections.

## [5.](#) Two Levels of Store-And-Forward

Basic email transfer is accomplished with an asynchronous store-and-forward communication infrastructure. This means that moving a message from an originator to a recipient involves a sequence of independent transmissions through some number of intermediaries, called MTAs. A very different task is the user-level process of re-posting a message through a new submission process, after final delivery for an earlier transfer sequence. Such MUA-based re-posting shares some functionality with basic MTA relaying, but it enjoys a degree of freedom with both addressing and content that is not available to MTAs.

The primary "routing" mechanism for Internet mail is the DNS MX record [[RFC1035](#)]. It is an advertisement, by a recipient domain, of hosts that are able to relay mail to it, within the portion of the Internet served by this instance of the DNS.

### [5.1](#) MTA Relaying

MTAs relay mail. They are like packet-switches and IP routers. Their job is to make routing assessments and to move the message payload data closer to the recipient. It is not their job to reformulate the payload or to change addresses in the envelope or the content.

### [5.2](#) MUA Forwarding

Forwarding is performed by MUAs that take a received message and submit it back to the transfer service, for delivery to one or more different addresses. A forwarded message may appear identical to a relayed message, such as for Alias forwarders, or it may have minimal similarity, as with a Reply.

#### [5.2.1](#) MUA Basic Forwarding

The simplest type of forwarding involves creating an entirely new

message, with new content, that includes the original message between



Originator-1 and Recipient-1. However this forwarded communication is between Recipient-1 (who could also be called Originator-2) and a new recipient, Recipient-2. The forwarded message is therefore independent of the original message exchange and creates a new message dialogue.

### [5.2.2](#) MUA Re-Sending

A recipient may wish to declare that an alternate addressee should take on responsibility for a message, or otherwise become involved in the original communication. They do this through a user-level forwarding function, called re-sending. The act of re-sending, or re-directing, splices a communication between Originator-1 and Recipient-1, to become a communication between Originator-1 and new Recipient-2. In this case, the content of the new message is the old message, including preservation of the essential aspects of the original message's origination information.

Identities specified in a resent message include

Identity	Role
<a href="#">RFC2822</a> .From	Names and email addresses for the original author(s) of the message content are retained. The free-form (display-name) portion of the address might be modified to provide informal reference to the agent responsible for the redirection.
<a href="#">RFC2822</a> .Reply-To	If this field is present in the original message, it should be retained in the Re-sent message.
<a href="#">RFC2822</a> .Sender	This field is expected to contain the original Sender value.
<a href="#">RFC2822</a> .TO, <a href="#">RFC2822</a> .CC, <a href="#">RFC2822</a> .BCC	These specify the original message recipients.
<a href="#">RFC2822</a> .Resent-From	The address of the original recipient who is redirecting the message. Otherwise, the same rules apply for the Resent-From field as for an original <a href="#">RFC2822</a> .From field

<a href="#">RFC2822</a> .Resent-Sender	The address of the agent responsible for re-submitting the message. For efficiency, this field should be omitted if it contains the same address as <a href="#">RFC2822</a> .Resent-From. However this does not mean there is no Resend-Sender specified. Rather, it means that that header is virtual and that the address in the Resent-From field must be used. Specification of the error return addresses (the "bounces" address, contained in <a href="#">RFC2821</a> .MailFrom) is made by the Resent-Sender. Typically the bounce address is the same as the Resent-Sender address. However some usage scenarios require it to be different.
<a href="#">RFC2822</a> .Resent-To, <a href="#">RFC2822</a> .Resent-cc, <a href="#">RFC2822</a> .Resent-bcc	The addresses of the new recipients who will now be able to reply to the original author.
<a href="#">RFC2821</a> .MailFrom	The agent responsible for re-submission ( <a href="#">RFC2822</a> .Resent-Sender) is also responsible for specifying the new <a href="#">RFC2821</a> .MailFrom address.
<a href="#">RFC2821</a> .Rcpt-to	This will contain the address of a new recipient
<a href="#">RFC2822</a> .Received	When re-sending a message, the submission agent may record a Received header, to indicate the transition from original posting to resubmission.

Table 5: ReSent Identities

### [5.2.3](#) MUA Reply

When a recipient formulates a response to a message, the new message is not typically viewed as being a "forwarding" of the original.

Internet-Draft

MailArch

May 2004

#### [5.2.4](#) MUA Gateways

Gateways perform the basic routing and transfer work of message relaying, but they also make any message or address modifications that are needed to send the message into the next messaging environment. When a gateway connects two differing messaging services, its role is easy to identify and understand. When it connects environments that have technical similarity, but may have significant administrative differences, it is easy to think that a gateway is merely an MTA. The critical distinguish between an MTA and a gateway is that the latter modifies addresses and/or message content.

A gateway may set any identity field available to a regular MUA. Identities typically set by gateways include

Identity	Role
<a href="#">RFC2822</a> .From	Names and email addresses for the original author(s) of the message content are retained. As for all original addressing information in the message, the gateway may translate addresses in whatever way will allow them continue to be useful in the target environment.
<a href="#">RFC2822</a> .Reply-To	The gateway should retain this information, if it is originally present. The ability to perform a successful reply by a gatewayed recipient is a typical test of gateway functionality.
<a href="#">RFC2822</a> .Sender	This may retain the original value or may be set to a new address
<a href="#">RFC2822</a> .TO, <a href="#">RFC2822</a> .CC,	These usually retain their

<a href="#">RFC2822</a> .BCC	original addresses.	
<a href="#">RFC2821</a> .MailFrom	The agent responsible for	
	gatewaying the message may	
	choose to specify a new address	
	to receive handling notices.	
<a href="#">RFC2822</a> .Received	The gateway may record a	
	Received header, to indicate	
	the transition from original	
	posting to the new messaging	

Crocker

Expires November 7, 2004

[Page 17]

Internet-Draft

MailArch

May 2004

	environment.	
+-----+	+-----+	+-----+

Table 6: Gateway Identities

#### [5.2.5](#) MUA Alias Handling

A simple re-addressing facility that is available in most MDA implementations is called Aliasing. It is performed just before placing a message into the specified recipient's inbox. Instead, the message is submitted back to the transfer service, for delivery to one or more alternate addresses. Although implemented as part of the message delivery service, this facility is strictly a recipient user function. In effect it resubmits the message to a new address, on behalf of the listed recipient.

What is most distinctive about this forwarding mechanism is how closely it compares to normal MTA store-and-forward. In reality its only interesting difference is that it changes the [RFC2821](#).RCPT-TO value. Notably it does not typically change the [RFC2821](#).Mailfrom

An MDA that is re-posting a message to an alias typically changes only envelope information:

+-----+	+-----+	+-----+
Identity	Role	
+-----+	+-----+	+-----+
<a href="#">RFC2822</a> .TO, <a href="#">RFC2822</a> .CC,	These retain their original	
<a href="#">RFC2822</a> .BCC	addresses.	
<a href="#">RFC2821</a> .Rcpt-To	This field contains an alias	
	address.	

<a href="#">RFC2821</a> .MailFrom	The agent responsible for submission to an alias address will usually retain the original address to receive handling notifications. The benefit of retaining the original MailFrom value is to ensure that the origination-side agent knows of that there has been a delivery problem. On the other hand, the responsibility for the problem usually lies with the recipient, since the Alias mechanism is strictly under the recipient's control.
-----------------------------------	---

Crocker

Expires November 7, 2004

[Page 18]

Internet-Draft

MailArch

May 2004

<a href="#">RFC2821</a> .Received	The agent should record Received information, to indicate the delivery to the original address and submission to the alias address. The trace of Received headers should include everything from original posting through final delivery to the alias.
-----------------------------------	--

Table 7: Alias Identities

#### [5.2.6](#) MUA Mailing Lists

Mailing lists have explicit email addresses and forward messages to a list of subscribed members. Mailing list processing is a user-level activity, outside of the core email transfer service. The mailing list address is, therefore, associated with a distinct user-level entity that can perform arbitrary actions upon the original message, before submitting it to the mailing list membership. Hence, mailing lists are similar to gateways.

Identities set by a mailing list processor, when submitting a

message, include:

Identity	Role
<a href="#">RFC2919</a> .List-id	This provides a global mailing list naming framework that is independent of particular hosts. Although <a href="#">RFC2919</a> is a standards-track specification, it has not gained significant adoption.
<a href="#">RFC2369</a> .List-*	<a href="#">RFC2369</a> defines a collection of message headers for use by mailing lists. In effect, they supply list-specific parameters for common mailing list user operations. The identifiers for these operations are for the list, itself, and the user-as-subscriber.
<a href="#">RFC2822</a> .From	Names and email addresses for the original author(s) of the

<a href="#">RFC2822</a> .Reply-To	message content are specified. Mailing lists have introduced an ambiguity for the Reply-To field. Some List operations choose to force all replies to go to all list members. They achieve this by placing the list address into the <a href="#">RFC2822</a> .Reply-To field. Hence, direct, "private" replies only to the original author cannot be achieved by using the MUA's typical "reply to author" function. If the author created a Reply-To field, its information is lost.
<a href="#">RFC2822</a> .Sender	This will usually specify the address of the agent

[RFC2822](#).TO, [RFC2822](#).CC

[RFC2821](#).MailFrom

[RFC2821](#).Rcpt-To

[RFC2821](#).Received

responsible for mailing list operations. However, some mailing lists operate in a manner very similar to a simple MTA relay, so that they preserve as much of the original handling information as possible, including the original [RFC2822](#).Sender field. These will usually contain the original list of recipient addresses.

This may contain the original address to be notified of transmission issues, or the mailing list agent may set it to contain a new notification address. Typically, the value is set to a new address, so that mailing list members and posters are not burdened with transmission-related notifications.

This contain the address of a mailing list member.

An Mailing List Agent should record a Received header, to indicate the transition from original posting to mailing list forwarding. The Agent may

choose to have the message retain the original set of Received headers or may choose to remove them. In the latter case, it should ensure that the original Received headers are otherwise available, to ensure later accountability and diagnostic access to it.

Table 8: Mailing List Identities

## 6. Security Considerations

This document does not specify any new Internet mail functionality. Consequently it should introduce no new security considerations.

However its discussion of the roles and responsibilities for different mail service modules, and the information they create, highlights the considerable security considerations that must be present when implementing any component of the Internet mail service.

## 7 References

[I-D.hutzler-spamops]

Hutzler, C., Crocker, D., Resnick, P., Sanderson, R. and E. Allman, "Email Submission Between Independent Networks", [draft-hutzler-spamops-00](#) (work in progress), March 2004.

[ID-HDR-Reg]

"Registration of mail and MIME header fields", [draft-klyne-hdrreg-mail-04.txt](#) (work in progress), Apr 2004.

[ID-SPAMOPS]

Hutzler, C., Crocker, D., Resnick, P., Sanders, R. and E. Allman, "Email Submission Between Independent Networks", I-D [draft-hutzler-spamops-00.txt](#), March 2004.

[RFC0821] Postel, J., "Simple Mail Transfer Protocol", STD 10, [RFC 821](#), August 1982.

[RFC0822] Crocker, D., "Standard for the format of ARPA Internet text messages", STD 11, [RFC 822](#), August 1982.

Crocker

Expires November 7, 2004

[Page 21]

---

Internet-Draft

MailArch

May 2004

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, [RFC 1034](#), November 1987.

[RFC1035] Mockapetris, P., "Domain names - implementation and



- specification", STD 13, [RFC 1035](#), November 1987.
- [RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, [RFC 1939](#), May 1996.
- [RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", [RFC 2045](#), November 1996.
- [RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", [RFC 2046](#), November 1996.
- [RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", [RFC 2047](#), November 1996.
- [RFC2048] Freed, N., Klensin, J. and J. Postel, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", [BCP 13](#), [RFC 2048](#), November 1996.
- [RFC2049] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples", [RFC 2049](#), November 1996.
- [RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", [RFC 2181](#), July 1997.
- [RFC2304] Allocchio, C., "Minimal FAX address format in Internet Mail", [RFC 2304](#), March 1998.
- [RFC2369] Neufeld, G. and J. Baer, "The Use of URLs as Meta-Syntax for Core Mail List Commands and their Transport through Message Header Fields", [RFC 2369](#), July 1998.
- [RFC2421] Vaudreuil, G. and G. Parsons, "Voice Profile for Internet Mail - version 2", [RFC 2421](#), September 1998.
- [RFC2423] Vaudreuil, G. and G. Parsons, "VPIM Voice Message MIME Sub-type Registration", [RFC 2423](#), September 1998.
- [RFC2442] "The Batch SMTP Media Type", [RFC 2442](#), November 1998.
- [RFC2476] Gellens, R. and J. Klensin, "Message Submission", RFC

2476, December 1998.

- [RFC2645] "On-Demand Mail Relay (ODMR) SMTP with Dynamic IP Addresses", [RFC 2465](#), August 1999.
- [RFC2782] Gulbrandsen, A., Vixie, P. and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2782](#), February 2000.
- [RFC2821] Klensin, J., "Simple Mail Transfer Protocol", [RFC 2821](#), April 2001.
- [RFC2822] Resnick, P., "Internet Message Format", [RFC 2822](#), April 2001.
- [RFC2919] Chandhok, R. and G. Wenger, "List-Id: A Structured Field and Namespace for the Identification of Mailing Lists", [RFC 2919](#), March 2001.
- [RFC3501] Crispin, M., "Internet Message Access Protocol - Version 4rev1", [RFC 3501](#), March 2003.

#### Author's Address

Dave Crocker  
Brandenburg InternetWorking  
675 Spruce Drive  
Sunnyvale, CA 94086  
USA

Phone: +1.408.246.8253  
EMail: [dcrocker@brandenburg.com](mailto:dcrocker@brandenburg.com)

#### [Appendix A](#). Acknowledgements

The Email Architecture section derives from [draft-hutzler-spamops \[ID-SPAMOPS\]](#). The text has been further elaborated.

Graham Klyne, Pete Resnick and Steve Atkins provided thoughtful insight on the framework and details of early drafts.

Internet-Draft

MailArch

May 2004

## Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#). Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

## Full Copyright Statement

Copyright (C) The Internet Society (2004). 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 organizations, 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 assignees.

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

Crocker

Expires November 7, 2004

[Page 24]

---

Internet-Draft

MailArch

May 2004

HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

#### Acknowledgment

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

Crocker

Expires November 7, 2004

[Page 25]