Internet Working Group
Internet Draft
Document: draft-fakih-amdp-00.txt
Category: Standards Track
Expires: August 2003

A. El Fakih

January 2003

Adaptive Mail Delivery Protocol (AMDP)

Status of this Memo

Abstract

This document describes the Adaptive Mail Delivery Protocol (AMDP). It aims to resolve the problems associated with current email systems that rely on the mail delivery process defined by the Simple Mail Transfer Protocol (SMTP). This is done by extending and designing a backwards-compatible replacement for SMTP, as well as restructuring the mail delivery process. The process is built around an adaptive scheme that is able to addresses current and future demands for a secure and reliable e-mail delivery systems.

Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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-2119 [ii].

The symbols -> and <- are used to indicate main flow of an email message and its direction, where -> means email flows from the object on the left to the one on the right, while <- is the flow in the reverse direction. <-> indicates that a continuous two way socket connection is used to exchange information, while <=> indicates that separate socket connections are required to complete the transaction.

In the document the following numerals are used to reference various stages of the mail delivery:

Number	Stage ownership	Description
10	Sender	E-mail Client
15	Sender	Mail Authentication Service (MAS)
20	Sender	Outgoing Mail Service (OMG)
30	Sender	SMTP outgoing mail gateway
40		DNS or MHF authentication Service
50	Sender	Mail Holding Facility Service (MHF)
60	Receiver	Mail Information Service (MIS)
65	Receiver	Subscription service
70	Receiver	Incoming Mail Gateway Service (IMG)
80	Receiver	Mail Storage Service
90	Receiver	E-mail Client
110	External	Mailing list management services
120	External	Mail abuse network

Table of Contents

1. Introduction

The Adaptive Mail Delivery Protocol (AMDP) is an ambitious project that aims to solve the problems associated with the current mail delivery system. This is done by extending and designing a backwards-compatible replacement for the Simple Mail Transfer Protocol (SMTP), as well as restructuring the mail delivery process. AMDP is built around an adaptive scheme that is able to addresses current and future demands for a secure, and reliable e-mail delivery systems.

Current SMTP based mail systems, suffer from many serious and costly to fix issues. These issues include impersonating others, lack of privacy, virus spread, spamming, among others. This is made possible due to the inherent design of SMTP, which was never designed to be used the way we use it today. AMDP is designed to address and solve these issues, while allowing in its design the flexibility and adaptability required to grow with the needs of the Internet.

Current technical solutions designed to curb the continuous abuse of electronic mail systems, are ineffective in the long run due to the inherent design of SMTP and the mail delivery process in general.

The continual rise of electronic mail abuse is bound to have a negative effect on the growth and progress of businesses utilizing electronic mail as a business tool. Therefore it is imperative that we address and solve the issues plaguing the mail delivery process, allowing us to have a better tool for conducting business, promoting education and entertainment within a safe environment. AMDP aims to achieve the following:

Prevent and Control Abuse

Most of the Internet abuse taking place is due to the early designs of network protocol with relied heavily on implicit trust between networked systems. However with the proliferation of the Internet and its wide use on a global scale, this trust has been abused by individuals for profit and fun, which made controlling the various types of abuse difficult and costly. Most of the solutions geared to deal with the growing number of abuse tend to band-aid the problem by filtering received email massages after it is received by the incoming mail server. The messages are filtered using sophisticated programs to control SPAM, Viruses, among others. However none has tried to solve the problem from its roots, and AMDP has been designed from the ground up to deal with these issues.

In AMDP only an envelope is accepted, and only after a routine authentication takes place that does not require any encryption, the message and/or envelope can be accepted for delivery.

Control and allow Unsolicited Mail

Everyone using the internet has dealt with unsolicited mail in one form or another. Unsolicited mail, also referred to as SPAM, had gone from being an annoyance to become a major cost for doing business on the net.

After technical solutions have failed to control the wide abuse of email systems, legislations in the US and Europe have tried to control it, but to no avail. Tracking spammers is impossible in most cases and expensive both from a technical and logistical points-of-view. It is sad to see that companies have been set up to explicitly abuse these email vulnerabilities, claiming to abide by the law while they themselves indulge in the profitable business of spamming users on the expense of others.

For the reader that wonders how it is possible, the answer is straightforward. Any number of unsolicited messages can be sent from any unprotected host to any user whether they want it or not. The message can claim itself to be from any person or any organization whatsoever. System administrators do not have the tools to stop the abuse, and are unable to keep up with the various permutations used to bypass mail filters through protected systems. The process of protecting email systems from this kind of abuse is estimated to cost companies millions of dollars in damages every year.

Although many have negative feelings associated with the practice of mass unsolicited mail, many users benefit from these services if they are not abused. There are many legitimate uses of mass mailing and should be allowed to exist within a controlled environment that is marketing friendly, without trampling all over the rights of the recipient for privacy, or for the business rules of the network provider, and the companies owning the network infrastructure.

AMDP is designed to be friendly to businesses dealing with mass mailing, while providing within the process various control mechanisms for the end recipient and the domain name administrators as to what kind of mail they want to receive, and in what frequency.

Control the spread of Viruses

One of the side effects of the unauthenticated mail transports is the ease in which viruses are spread all over the Internet. During a virus attack (Virus storms) millions of users receive viruses from people who may or may not know, who had their computer systems compromised. Other attacks end up sending the user's private files. Viruses tend to spread from one user to another without any means of control. And even after an attack has been identified, it is very difficult for mail administrators to control the flow of these viruses.

This can be controlled, and even prevented if the identity of the sender is truly known to the user, and there are tools the service provider can use to prevent its mail system to be hijacked by the virus.

Authenticate sender and receiver

SMTP does not have any means to authenticate the address of the sender, nor does it authenticate the claims made by a server that is sending mail on behalf a domain name. The authentication is

assumed to be done on one side of the mail delivery scheme, and the receiving side has to trust the sender's address in the mail envelope. This is a very poor design that is the heart of many of the abuse cases related to email. Basically SMTP allows anyone to claim to be anyone they choose to.

AMDP addresses these points in a way that is both easy to implement, manage, and it is very costly/difficult for spammers to overcome.

Integrate Encryption for better Security

Current systems rely on third party software to encrypt messages on the client side. Encryption usage is very limited due to the perceived complexity of setting up mail applications, or not understanding how encryption works and how to encrypt and decrypt their messages. AMDP allows for automatic negotiation for general encryption between mail servers, making it difficult for persons who capture the mail message to read the content without having to decrypt it.

Multilingual support

AMDP supports language negotiation, where the language of the message is identified for possible automatic translation. It also aims to use Unicode as its method of communication leveling the ground for the various encoding existing today.

Native language domain name support

AMDP will support non-Latin based domain names.

Subject classification support

AMDP supports the classification of email messages. The messages can be tagged as business, adult, personal, marketing, etc.. The unified classification will give parents/service providers some control over adult mail being delivered to minorsÆ mail boxes. Although this system may seem to require implicit trust, intelligent parsers, in combination to the business of classification certificates will be used for this purpose.

Separate business and information technology mail delivery rules

SMTP mail systems do not allow businesses to decide independently from the Information Technology (IT) department what mail should enter its systems, at what hours, volume, or set the priority for

inbound and outbound processing. AMDP is designed to take these into consideration and allows the business to make its own rule separately from IT to decide what goes in, what goes out, and when. While IT has its own rules such as what types of mail is allowed, quota's etc..

AMDP also introduces the concept of public and private mail delivery rules. Where the public rules are available for mail servers to parse before attempting delivery, minimizing the amount of wasted bandwidth due to the guaranteed rejection of any mail message meeting the criteria.

Streamlined law enforcement and dispute resolution

Authenticated systems, can give law enforcement a better tool to track effectively messages to its source. While classifications and automatic error reporting can lead to better control of abuse and talking legal actions (if necessary) against systems that do continue to abuse the mail system. However the intention of the design is to make it unprofitable to abuse mail systems.

Better support for white and black lists

Current methods rely on black or white lists to ban or allow mail servers to deliver mail. These lists are not easy to use, maintain, register or remove a host from it. They are prone to block people who are not involved in Spam, or allow others who do. Hence AMDP will have a standard way to report these abuses and include specific information required to authenticate an abuse has occurred, and the severity of the abuse.

Guaranteed mail delivery with return receipt

AMDP design allows the email system to be used as a method to guarantee delivery of electronic deliverables. Users purchasing software, images, music, or any other media that can be digitized can receive the actual deliverable into their mailbox. It stays at the sellerÆs servers, until it is requested by the user, which in turn executes or finalizes the sales transaction by verifying that the deliverable was received.

Current solutions using similar approach do not guarantee receipt of a message.

Electronic postage support

AMDP design incorporates the option to accept payment for electronic mail received. The payment can be perceived as electronic postage, and it is set by the domain name

administrators. Mail messages meeting the requirements, and pay the postage are accepted for delivery.

This option is also used as a natural method to control the growth of unsolicited electronic mail, since these messages will have to pay for access, which makes it unprofitable for companies to mass mail non interested users.

2. Current electronic mail delivery process

Mail delivery over electronic networks, such as the Internet, relies heavily on an early design defined in the Simple Mail Transfer Protocol (SMTP) specifications RFC 821 iii. The SMTP specifications outlined a simple approach to transmit a message from a host computer to a recipient mail server. Figure 1 illustrates the path of a valid email message starting from the initiating user [10] to its final destination [90].

[10] -> [30] -> [70] -> [80] <-> [90]

Figure 1. SMTP mail delivery process

Note: The numerals used in Figure 1 are explained in the 'Conventions section' of this document.

To understand the current mail delivery process we should trace the journey of an email message to its destination.

1. A mail message starts its journey from a personal computer [10], where the user types in his/her message into an e-mail client such as Outlook on Windows platform, or Pine on the Unix platform,. The e-mail client will then assemble together the text of the e-mail and the information required for delivery such as sender and receiver emails, date, etc. into a standard format that can be read by SMTP based mail servers, and forward the email to the assigned outbound SMTP server [30].

2. The message is received, the message header is read, the recipient e-mail is extracted from it, a domain name lookup is done using the DNS protocol [40] to determine the address of the server assigned as a the mail gateway for the domain in questions, and then the message is forwarded to this server [70] using the SMTP protocol.

Note: The address of the mail server designated as mail gateway [70] is publicly available through the DNS protocol under the MX record for any given domain name.

3. The receiving mail server [70] checks if the user exists, cross references the email with known black lists, or matches it against internal rules, then it accepts the message, and responds to the sender server [30] that it has been accepted. Optionally the receiving server would check for viruses, available space, or uses other filtering techniques to determine if the message is unsolicited, or blacklisted. If it fails any of these tests it will reply with an email message with the error to the senderÆs email defined in the header of the message itself. However if the message passes the test it is forwarded to a local or remote delivery agent.

4. The delivery agent will then save the message into the mailbox of the recipient [80] where it will reside until it is retrieved by the recipient using IMAP, POP or other methods using an email client [90].

5. Once the recipient opens his e-mail client [90], all new mail is retrieved from the mail server [80], and displayed as a list showing the subject, recipients, and dates of each message. The user can then proceed to read, delete, or respond to any message his/her mailbox.

2.1 Problems with current system

The current process, described in the previous section, relies heavily on the content of the e-mail message to decide how to route the message. It does not verify the accuracy of any of the information contained within the message. Like many of the early protocol designs of the Internet, these systems were designed with an implied trust built into them. This trust is exploited for fun, profit, or as an act of aggression.

Some of the problems plaguing the current mail delivery process include:

1. The inability to truly authenticate the identity of sender or server relaying the mail message. These two important factors are the building block for the practice of unsolicited mail.

2. The inability to adequately control the spread of unsolicited or virus-packed mail at either side of the mail delivery process.

3. The inability to isolate, combat and defend from denial of service (DoS) attacks that rely on open relay gateways, and forged return addresses.

4. The inability to reliably track the delivery or non-delivery status of a given message.

5. The inability to control the type, size, or language used in outgoing mail by system administrators.

6. The inability of domain name owners to prevent others from utilizing their domain in attacks, or unacceptable business practices.

7. Absence of encryption in the design of SMTP allows anyone to freely intercept and read other peopleÆs mail, making it a poor instrument to exchange sensitive information without accepting the risks associated with such system.

2.2 Scenarios of email abuse

Listed herein are few scenarios of electronic mail abuse due to the loose design of the SMTP mail delivery process.

Forged Mail:

A user connects to an SMTP server (This could be done manually by connecting using telnet to port 25, or via an available program designed for that purpose). The user will then provide the SMTP server with a forged sender address (FROM), a valid recipient (RCPT TO), and the text of the message to send (DATA). The recipientÆs SMTP server accepts the message, since it has no tools to authenticate the information provided related to the sender. The recipient becomes a victim of a forged message, which is the building block of all mail based attacks.

Place blame on someone else:

This is a variation of the "forged mailö case, however in this one the user sends offensive messages, adult material, viruses, or tries to sell something. The sender provides a valid return address which does not belong to him, and that is not on any known black lists. The owners of the domain or email account are subsequently blamed for the unsolicited mail, placed on black mailing lists, and have to deal with angry messages, and communicate with the various black list organizations to attempt and remove their domain from the lists, while they are in fact victims of the attack as well.

Denial of Service Attacks:

In this case, the attacker is interested in inflicting damage to one or more parties. The attacker starts by selecting their victimsÆ email addresses, which would be used as return address. The attacker proceeds to send mail to million of servers all over the net targeting existing and non-existing email accounts. Error messages, requests for removal, and replies would be directed to the victimsÆ accounts. The ISP servicing the accounts and their users will have to abandon the accounts, unless they have the time and money to defend against the variations of the message, and servers responding to the message etc.

Another type of attack is to send large attachments to victim accounts which would fill the mailbox of the recipient, and cripple the ISP if they are not equipped to deal spam, or the mail volume generated by this kind of direct attack.

While other attacks rely on relay mail servers which are used to attack one or more targets by using thousands of hosts to email the same target at the same time, and for extend period of time bringing the mail server to a complete stop.

Remove email from lists:

There are many schemes out there designed to obtain email addresses against their ownerÆs permission for the purpose of Spam. These schemes include mass mailing and asking people to unsubscribe. Unknown to the victim, by attempting to unsubscribe, they validate the email account for further Spam. Other schemes include harvesting web pages for email addresses, emailing them, and checking for error messages. Email addresses that do not generate errors are kept for future Spam campaigns. SMTP itself has a design flaw that helps in the chaos. SMTP has a command called (VRFY) it was designed to verify the existence of username on the email system contacted, which was abused and used as a tool to verify emails for Spam. Today many SMTP servers block this feature.

3. Proposed electronic mail delivery process

3.1 The Philosophy

The proposed mail delivery scheme revolves around the concept of shifting the bulk of responsibility of storing and delivering the message away from the receiver and onto the sender contrary to the way it is currently implemented.

The current email delivery scheme places the burden of processing, storing, and delivering mail messages on to the recipient side. The process is prone to abuse, allowing any user equipped with a list of emails and open relay servers to send millions of unsolicited messages creating havoc at the recipient side. This is due to absence of mutual responsibility in the mail delivery cycle. In most cases mass mailing contain jokes, viruses, chain letters, or marketing material which require a minimum investment from the senderÆs side, while the recipient must make a larger investment in time and money to process and store unsolicited mail.

The target of AMDP is it to make the sender responsible for his actions. As in our everyday life, the sender will store the message on his own server, notify the recipient of the existence of a message on the server, and serve the email message when requested.

This is similar to what happens when you ship something via a commercial carrier be it the local post office, UPS, DHL, or other courier services. The sender is responsible for the delivery of the message whether it is done directly, or via an agent. His responsibility ends when the person receives or rejects the message.

Historically, we have used mail schemes that placed the burden of delivering the message on the sender, and we know it works, and with AMDP we borrowed these tested concepts to design an online system that is adaptive and can operate safely within the Internet today and in the future.

3.2 AMDP design stages and implementations

This design document outlines the steps that have to be accomplished while transitioning existing software that interacts with the mail delivery scheme from its current state to a system that supports all features of AMDP explicitly.

The design document will refer to the transitional and final stages where:

The transitional stage:

> In this stage, SMTP, DNS and AMDP systems coexist as they are. Without this stage, the system is of no use to anyone. The system will have to use SMTP readable email envelopes to allow old systems to send, and receive messages to, and from AMDP servers. The system will have to rely on current DNS structure for its MHF authentication, and use available technologies such as secure shell to encrypt communications between venerable points. It will also use HTTP to retrieve messages where applicable.
>
> Typically an envelope will contains both the SMTP and AMDP header in all of it communications, however this can be dynamically determined when an MHF [50] connects to a mail gateway [70] and decides if it supports AMDP or SMTP based on the HELO command, which could be changed to receive a variation of the command to identify AMDP servers.

The final stage:

> In this stage, SMTP, DNS, POP and AMDP systems coexist but changes to their architecture has been completed, and refined. Other parts of the scheme that need to be in place include authentication mechanisms of AMDP, mail classification, reporting of abuse, postage micro payments, etc.. A modified version of POP may be needed to deliver messages to their ultimate destination, unless HTTP or other specially designed protocol is used for that purpose.

This paves the way for a design that can be enforced technically and legally. Users continuing to use SMTP can coexist with AMDP, however they will be treated as bulk mail sources, and would be isolated in the long run. An analogy of SMTP and AMDP would be Archie or Gopher protocols versus http, where they took a back seat as http gained support because it delivered convenience and results compared to its predecessors.

The simplest way to explain how AMDP works is to follow an e-mail message from sender to recipient, and see how the message is treated on its way to the final destination (See Figure 2). The following sections will outline the AMDP mail delivery process as to give the reader an overview of the process, followed by a revisit to the process with an in-depth approach explaining what occurs at each stage both in the transitional and the final stages of the design.

4. Outline of AMDP delivery process

AMDP relies on the same client-server communication process used in SMTP. i.e. it will use the basic MAIL FROM, RCPT TO, and DATA to transfer data from one server to another. However the major change is in the mail envelope itself.

This section outlines the path traversed by a valid email message traveling from [10] to [90] shown in Figure 2. The details of what happens at each step are discussed in section 5 ..

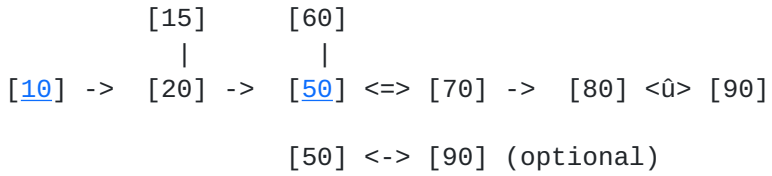

Figure 2. AMDP mail delivery process

1. A user composes a message and sends the message from an e-mail client [10].

2. The message is then received by the email gateway [20] (OMG) where it will undergo various business tests, header information is added to the email envelope, and it is forwarded to the Mail Handling Facility (MHF) [50].

The mail holding facility (MHF) is the location where outgoing mail is held until it is delivered to the recipient [90]. MHF also keeps track of delivery status of any email message, making it possible to execute e-commerce transactions using the MHF to guarantee delivery.

3. The mail holding facility (MHF) [50] contacts the mail information server (MIS) [60] and analyzes the public mail policy available online. If the public policy does not deny mail from 50, then it will create a standard envelope using AMDP defined headers, and send it to the appropriate mail server for further processing [70].

4. The recipient server [70], also referred to as Incoming mail gateway (IMG), will read the incoming envelope, authenticate the MHF, and if the envelope meets their business requirements, it is forwarded to the user mailbox [80].

5. The messages or envelopes are stored in the server [80] until they are accessed by the final recipient.

6. The recipient [90] will use an email client to retrieve the available envelopes in his/her mailbox. Once the user decides to read the message, it is retrieved from [80] if it was accepted by [70] in its entirety, or it has to be retrieved from the MHF [50] using any of the available transports such as HTTP, IMAP, POP or any other protocol designed for this purpose.
If the MHF is not online, the client can be programmed to poll the MHF at various intervals until a connection is made, or request from IMG [70] to poll the data from the MHF [50] and deliver the message to the mail storage [80].

5. Details of AMDP delivery process

AMDP relies on the same client-server communication process used in SMTP. i.e. it will use the basic MAIL FROM, RCPT TO, and DATA to transfer data from one server to another. However the major change is in the mail envelope itself.

In this section we will explain in detail the delivery process, and some of the possible variations. The specification may include separate sections for the transitional and final implementation phases.

1. A user composes a message and sends the message from an e-mail client [10].

   During the transitional stage, no changes are required to email clients. All outgoing email should work the same.

   However in the final stage, the email client would provide key information required by MHF [50], such as the mail classification, language and encoding of message. The program should also have better error reporting interface to understand why an outgoing message failed. It will also have better mechanisms in resolving cryptic error message generated by SMTP that most users do not understand, especially when we talk about non-English speaking users, and instruct them on what steps to take to remedy the situation. The language setting of the mail client will enable the mail gateway [20] to deliver the correct error message. Email clients must be able to read AMDP generated envelopes and retrieve the message automatically, instead of using a separate web browser, it will also allow for MHF polling, etc..

2. The message is then received by the outgoing mail gateway (OMG) [20] where it will undergo various business tests.

The server [20] MUST

Use a trusted connection between [10] and [20]. This can be achieved by enforcing the use of assigned internal IPÆs, firewall, encryption, etc. It is also recommended that the connection does not use a clear text mechanism when possible.

Use a username and password to authenticate the user when accepting outgoing mail, by checking the Mail authentication server MAS [15].

Replace the AMDP-FROM: field of the AMDP envelope with the proper email address of the user. This prevents common mistakes made by new users of the Internet, as well as deliberate forgery of senderÆs information.

Add other header information required by MHF such as language, classification, etc.

Optionally server [20] can:

Check for bad language, scan for viruses, enforce outbound file size limits, as well as computer quota restrictions for outgoing mail. The server can also block users from sending messages for non-payment, parental control setting, or due to previous history of email abuse.

If the message is refused for outbound delivery for any reason, the user will be informed (preferably via a direct method, to eliminate any chance of flooding a userÆs mail account with error reports as it happens in SMTP) with the reasons for denying the message, and the actions which need to be taken by the sender to remedy the situation.

However if the message satisfies the business rules, other header information are added to the original mail message envelope, and the message is forwarded to the Mail Holding Facility for delivery [50].

3. The mail holding facility (MHF) 50 is responsible for holding all outgoing mail for delivery. The MHF plays an important role in the mail delivery cycle which includes:

a. Checking public mail policies prior to attempting delivery.

b. Composing standard AMDP envelopes.

c. Sending notification of mail to recipients.

d. Holding the mail until it is picked up by the final recipient.

e. Forwarding the message to other severs that would accept messages instead of envelopes.

f. Keeping track of the delivery status of all email messages.

g. Providing a mechanism to report the delivery status, making it possible to conduct e-commerce transactions, by guaranteeing mail delivery status.

The mail holding facility has three or more possible configurations depending on the size of the email outgoing from its network.

Small size companies:

In this context, a small company generates a small number of outgoing messages. The administrator can use the same server that is currently used for processing outgoing mail [20], to act as the Mail Holding Facility [50], by dedicating some extra resources for the task.

Medium size companies:

In this context, a considerable amount of storage is needed for the outgoing mail. The administrator can opt to have a dedicated server with a larger amount of storage for the MHF [50] task.

Large size companies:

In this context, millions of messages are sent out for delivery (ISPs and mail service providers). The business would decide between having the outgoing mail handled internally using its own dedicated servers, or opt to outsource the task to specialized companies authorizing them to act as mail agents to deliver the mail.

Third party processing

In this context, a company specializes in processing mail for any party wishing to use an external service. This would act as a post office. In this configuration the third party becomes the authorized MHF for the domain name, and it can process all outgoing mail.

The MHF receives its messages from authorized outgoing mail servers (OMG) inside its network. The communication between servers [20] and [50] should be encrypted to protect it from receiving any forged information. However other methods of authentication can be used, as long as they explicitly have to accept connections from OMG, and deny all by default.

The MHF will then store and lock the message on the server, and issue a unique identifier (AMDP-MSG-ID) for the message.

If the message is intended for multiple users, the MHF will associate a different id for each one of the recipients.

The MHF will then communicate with the Mail Information Server (MIS) 60 to review the public mail policy. It can optionally access that information from its internal cache if the MIS information has not expired as specified by the MIS in previous queries made to the same domain.

If postage is required, then payment is made and the receipt number is attached to the header information of the envelope under AMDP-PAYMENT-RCPT.

If everything is acceptable, the MHF will then build a standard AMDP envelope that will be sent to the recipientÆs incoming mail server [70], i.e. the server identified in the DNS MX record of the domain.

The MHF server 50 MUST:

Compose a valid AMDP delivery slip, which is referred to as an envelope. The envelope can be read by either SMTP or AMDP servers.

The envelope will include at least the following information:

All required SMTP headers.

The sender's name and email address:
Syntax: AMDP-FROM-NAME, AMDP-FROM
The information is generated from the message received from the email gateway [20].

The recipient's name and email address
Syntax: AMDP-TO-NAME, AMDP-TO
This information is generated from the original email message [10], or outgoing mail gateway [20].

The subject of the letter
Syntax: SUBJECT
The information is generated from the original email message.

Mail classification of the letter content
Syntax: AMDP-MAIL-CLASS
Refer to the Mail classification section for the proposed structure.

List of attachments
Syntax: AMDP-ATTACHMENTS
It will include the standard list, size, units used in the size field, name, and type of attachments. This is required by the recipient servers to know what to expect, such as required space resources. This list will be matched by the receiving server/client to prevent the message from being altered in transit, or from erroneous reporting of message size.

Language of the message
Syntax: AMDP-LANGUAGE
The language flag is set to the ISO 639 code of the content language. It is increasingly important to define the language of a mail message. This flag will help the end user, or other pre-processing tools, to decide how to process the message. i.e. Will translation be required? Does the current platform support display of this language? etc. The language flag should be set by the sender [10], or mail gateway [20]. It can be overwritten by the mail holding facility [50].

Encoding of the message
Syntax: AMDP-ENCODING
The encoding of the message is important and need to be set as per RFC 2277 [iv]. Ideally the MHF should convert all messages it receives into a UTF8 during the transitional stage, and Unicode during the final stage.

Mail Holding Facility Name and ID
Syntax: AMDP-MHF-NAME, AMDP-MHF-ID
These identification strings identify the MHF to

receiving servers [70]. The AMDP-MHF-NAME is the domain

name of the MHF, while the AMDP-MHF-ID is a unique identifier for the MHF. There are two configurations for the AMDP-MHF-NAME and AMDP-MFH-ID

Message unique identification
Syntax: AMDP-MSG-ID
This is a unique id issued by the MHF [50] holding the message. The number will become one of the keys used to retrieve the message by the intended recipient. In the event a sender sends one message and copies five people. Each recipient will receive an envelope with a different MSG-ID. This id is an important tool is reducing Spam and the manner it is created should by chosen carefully as not to be predictable.

Expiration date
Syntax: AMDP-Expire-On
This is expiration date of a message. The MHF [50] tells the recipient how long the message will be kept in storage before it is removed. This is applicable to marketing materials that have a deadline, or newsletters etc.. These deadlines help the MHF to keep its data up-to-date, and to enable automatic removal of un-retrieved message.

Marketing campaigns, job opening, transaction, among other have deadlines, therefore all will benefit from a deadline after which their message is purged from the delivery cycle. The AMDP-Expire-On could be used as a key to authenticate a message between servers [50] and [70].

Timestamp
Syntax: AMDP-TIMESTAMP
This is simply the timestamp in UTC of when the envelope was created. This time stamp MUST be between the message's original time, and the delivery time. This key is used in the authentication process between [50]-[70] MHF-IMG, and [50]-[90] MHF-Final recipient.

Message Size
Syntax: AMDP-Size
This is the total size of the message, which includes the attachments as well. The AMDP-SIZE also specifies the unit of measure in the string.

MHF Authentication Port
Syntax: AMDP-Auth-Port
The MHF has to answer authentication queries from the recipients' incoming mail gateways [70], and hence it can specify the port on which it answers these queries.

Authentication keys
Syntax: AMDP-AUTH-KEYS
This is the list of authentication keys that need to be used by the IMG [70] when contacting the MHF [50]. If not specified then the key are assumed to be AMDP-TEMPSTAMP, AMDP-EXPIRE-ON, and AMDP-MSG-ID

Version Number
Syntax: AMDP-VERSION
This will include the version number of AMDP used in the envelope. This way servers can negotiate advanced commands as they become available.

Payment Receipt
Syntax: AMDP-PAYMENT-RCPT
This will inform receiving servers that postage has been made, and provide the information needed to retrieve the payment information.

Once the envelope is sent by the MHF [50] to the recipient mail server [70], the first task of the MHF is completed. The MHF will later have to authenticate the existence of the message, issue a confirmation number, serve the message, and notify the sender of the delivery status.

4. The AMDP recipient server [70] will accept envelopes that meet its business requirements, do not violate the public mail policy [60], and can authenticate themselves.

The process is presented herein as follows:

The server will receive the AMDP envelope, and while the connection is still open, return an OK code, then TERMINATE the network connection. Optionally the server checks if this is the first time it has been contacted by this MHF, and enforces a one envelope per unauthenticated MHF rule before it accepts future envelopes for processing. Once an MHF has passed this test, the server will accept further envelopes from the MHF.

The IMG server [70] proceeds to authenticate the MHF [50] by checking the IP of the network connection against the

allowed MHFÆs for the domain in the envelope [40]. It will also authenticate the enclosed AMDP-MHF-NAME, and or the AMDP-MHF-ID within the envelope. The connecting IP number of the MHF may be different from the one specified in the envelope as AMDP-MHF-NAME, however the IP MUST be explicitly allowed to send messages on behalf of the domain. For example Yahoo.com can have two sets of MHF servers, some that send notification envelopes, while others store and serve the mail messages. However in both instances these servers are authenticated as MHFs in the DNS entries of Yahoo.com.

Note: The authentication of the AMDP-MHF-NAME, and AMDP-MHF-ID are discussed in section 10.

When the recipient IMG [70] receives an envelope, it will cross check the email category against its public and private mail acceptance policy to decide if it is allowed to proceed to the internal mail servers. If the message is refused, no further action is taken, and the message will simply expire at the holding MHF, however it is possible for the server to report the incident to external incident reporting services [120] in the event the message was in violation of the public mail policy [60].

The receiving IMG [70] will contact the MHF [50], supply the message id, the timestamp of the envelope, and the expiration date of the message. The MHF within the same network session will reply with a true or false. If the response is True then it will reiterate the email address of the recipient, and issue a confirmation number. If the provided email address matches the one in the envelope, [70] will acknowledge the message, and hand the mail message to the mail delivery agent [80].

If at any of these steps, something fails, the envelope is dropped and no further actions are taken by IMG [70]. The reason behind not issuing any error messages is to protect MHFs from being victims of DoS attacks using forged envelopes. SMTP errors will be generated to non AMDP systems.

It is also possible that during this transaction, if the recipients IMG policy allows for direct receipt. i.e. the message sent is within its size limitations, and mail from the MHF is accepted, then the IMG can request from the MHF the message itself, which will be passed on to [80] instead

of passing the empty AMDP envelope.

5. Before we continue with the email at point [80], we need to check what happens at the mail holding facility [50].

This is the second task performed by MHF [50] once it is contacted by an IMG [70] with the correct message id, expiration date, and timestamp. It acknowledges the existence of the message and issues a confirmation number. It will respond negatively to any future requests using the same message id in any combination until the passing of the expiration date. Doing so will make it useless for an external source to try guessing ids for a message once it was acknowledged by the recipient server

The three pieces of information outlined herein will make it tough for an attacker to guess valid message ids, or to retrieve email addresses. And once the MHF has been queried and confirmed by the recipient server, the id can not be used to make any further queries. The sending MHF can increase the pieces of information it requires for authentication. In this example we used three keys; however the MHF can ask the receiving server to authenticate with more keys using the AMDP-AUTH-KEYS, of course with a reasonable maximum setting. The MHF can also limit itself to queries made by servers it already contacted and have not authenticated awaiting message ids.

The MHF will also know at this stage whether the message has been accepted for delivery and expect that the final user [90] to retrieve the message before the expiration date if they are available. These will be made available to the message reporting system.

Once the authentication has been successfully completed between [50] and [70], the MHF will unlock the message for reading by the end recipient. This implies that before the authentication process, the message can not be accessed because it is locked, and a confirmation number has not been issued.

The MHF also keeps track of the email topics also known by threads, by maintaining an active list of threads. The originating MHF will maintain the master copy of the thread index. When negotiating message ids, the servers can send the updated thread keys to the receiving server if it requires having the thread tree which is used to reference back the thread. This is useful to reduce the size of a message if it is a thread so you do not need to send the original message back and forth. A thread is also related to the to the classification scheme, where the originator or sender can

classify the message.

The thread information can also be used by the receiving mail server [70] to identify existing communications between two servers and allow mail to be transacted with lesser authentications between the two parties.

6. Once the message has passed all necessary authentications on the server side [70] a modified envelope is sent to the user mailbox [80], containing the original envelope along with the confirmation number, and other information required to retrieve the mail message by the end user. The message will then reside in the userÆs mailbox [80] until it is retrieved by the end user using POP or IMAP or other similar protocols.

   In the transitional stage:

   The message continues to be an SMTP mail message. It is readable by any email client. The message will have two parts, an SMTP header that includes the name, email, subject, date of the message, etc, and a body. The header will also include the AMDP headers. The body of the message will indicate that this is an envelope for a message which is being held at a given URL, it will tell the user the size, classification of the message, expiration time, etc.. A properly formed URL will automatically be highlighted by most mail clients, while in others cases the user can choose to cut and paste the special URL into a browser.

   In the final stage:

   The email message continues to be readable by regular mail clients, but it will carry the AMDP header information that will tell compliant email client to display the information differently. For example instead of showing a message that they have to click on, the message will be retrieved from the MHF using the MSG ID, confirmation number etc..

7. The delivery cycle is finished when the user opens his email client [90]. The email client will then connect to the IMAP or POP server [80] and retrieve the standard header information. The user will be able to see a list of available messages in the mailbox, along with its senderÆs name, size, category, date, expires etc..

   In the transitional stage:

   When the message is retrieved, they will receive the text message described in the previous section, the user will then proceed to click on the URL to open the content of the message.

In the final stage:

When the message is clicked, the email client will retrieve the message from the MHF using the HTTP, IMAP, POP or similar protocols.

However in both cases the recipient must provide the MHF with the following information to retrieve the message. 1) The message ID 2) Expiration date 3)Time stamp, and 4) Confirmation number to be able to retrieve the message.

Once the message has been downloaded by the client, the client will use the same session to inform the MHF that the message has been delivered.

8. At this point the message has been successfully delivered by the MHF [50], and the following steps could be taken by [50] such as:

   a. Send a notification back to the sender using a simple envelope to the senderÆs mailbox [80-S] notifying them of the delivery status if it was requested, or make the information available through a web interface.

   b. Delete the message unless other ids are linked to the same message.

   c. Trigger an e-commerce transaction, such as issuing an invoice where the downloaded message could have been a software product, etc..

6. Private and Public Mail Acceptance Policy

Historically most mail servers such as sendmail, postfix, etc., allowed mail administrators to build private mail policies. This is done by installing and defining mail filters used to block unwanted, or virus packed mail.

AMDP introduces the concept of private and public mail policies. The private mail policy is used to further restrict mail wishing to be delivered to internal recipients, while the public policy is a method by which the recipient server publishes the rules by which it will accept mail for delivery [60]. Users wishing to email people within a given network are bound by these rules, or the mail will be ignored.

The MIS [60] serves all the information related to the public mail policy, being able to adapt to changing trends in the mail delivery process. The following is a sample MIS report, also referred to the public mail policy.

```
### START OF SAMPLE REPORT ########
DOMAIN: yahoo.com
MIS: mis.yahoo.com
SUPER-AUTH: auth0.yahoo.com
MAX-MAIL-SIZE: 200kb
MSG-PER-MIN: 60
DELIVERY-PRIORITY: [90AMDP] [10SMTP]
CONTACT-INFO: http://www.yahoo.com/contact/

# mail classification and rates
ACCEPT-CLASS: *::BULK::* [0.001]
ACCEPT-CLASS: *::BUSINESS::* [0.25]
ACCEPT-CLASS: *::GOV::* [1.00]
ACCEPT-CLASS: *::EDU::* [1.00]

#no charge for personal email
ACCEPT-CLASS: *::PERSONAL::* [0.00]

#charge $500 for unclassified mail
ACCEPT-CLASS: *::*::* [500.00]
DENY-CLASS: *::BULK::ADULT

ACCEPT-REJECT-PCT: 80%

# mail hours
UNSOLICITED_MAIL_HOURS: 18:00-21:00, 00:00-05:00

# Payment information
PAY: PAY.CENTIPAID.COM
PAY-METHOD: DIRECT
PAY-ACCT: YAHOO

#subscription servers
SUBSCRIBE-SYNCH: SUBSCRIBE.AYNA.COM:9012
SUBSCRIBE-AGENT: SUBSCRIBE.AYNA.COM:9012
### END OF SAMPLE REPORT ########
```

The public mail policy includes such items as:

Maximum mail size

It defines the maximum message size allowed for direct delivery. i.e. where a server allows the message to be accepted in its entirety instead of the envelope.

Number of mail per minute

It defines the maximum number of mail messages allowed from any domain name to be delivered within a minute. This is important for servers that are unable to process thousands of messages per minute when receive mail from large mailing lists. Mass mail software will tailor its speed to match the number to ensure that their messages will pass the mail gateway [70]

TBD: If [70] can reply to [50] with an ACCEPT-NEXT-IN field. The field will [50] how many minutes the MHF should have to wait before it attempts a new connection. If the value is set to 0 then the MHF is allowed to contact them immediately after.

Cache refresh rate

It defines the amount of hours before an MHF has to check the MIS for an updated copy of the public mail policy.

Delivery Priority Assignment

It states what is the delivery priority assigned to incoming mail message. For example a company may wish to assign 70% of its processing resources to AMDP compliant messages and 30% to SMTP based mail. This way each administrator can make public their level of tolerance of non-complaint mail servers.

Therefore an organization that will not accept any SMTP mail, it can setup the resource 100% AMDP 0% SMTP. The same model can be used to allow other types of protocols in the future.

Contact info

This defines a non-mail based form to communicate with the mail administrator (such as a URL, fax, phone, etc..). This is important for administrators of servers that are blocked from a given network, or for law enforcement agencies to contact the appropriate personnel about specific incidents using this method.

Rejected Classifications

This defines the classifications that are not accepted by the network administrator i.e. a government agency, or an elementary school do not want to accept any mail from marketing firms. And

any MHF contacting them with such messages will be reported back to the external incident reporting service [120].

Accept Classifications

This defines the classifications that are accepted by the network administrator.

Mail rates

Domain administrators can impose fees for accepting mail from certain mail categories, including all.

Payment information

If accepting postage for mail received, then all information related to the payment information, are included herein.

Unsolicited Mail hours (Universal Time)

It tells unsolicited mail providers what are the best times to deliver mail to the network. This is important for networks that wish to allow unsolicited mail to be accepted however within off peak times.

Subscription server (used in unsolicited mail delivery)

This server is used to help marketers build mailing lists that will be allowed through the mail gateway of a given domain. The subscribe server has two or more levels of subscriber acceptance. A user can accept to subscribe to a specific mail list, or to a mail classification.

The most popular case is when a company offers its users to signup to its mailing list. For the subscription to be accepted a few steps need to occur.

1. The external subscription [110] engine has to check the public policy [60] of the domain name, and if it does not find any problems (i.e. domain or classification banned) then it can automatically apply to be have an entry added to the server. The purpose for seeking this action, is to automatically build the required rules for the mailing list provider that will allow him/her access to the network.

2. The next step is done by the subscribe server [60], which will send a confirmation URL to the user in question [80], and

if the subscriber agrees, then user is added to the mailing list database in [60], and a message is sent to the originating subscription engine [110]. The subscribe server will contain similar rules to the ones use in mail gateway, such as one envelope rule, and deny any future requests if a specific number of requests are made from 110.

The sender must check the mailing list before trying to email any of the users on his list, to insure that their rejected/accepted message ratio is within the limit assigned by the domain admin.

The server is also used to synchronize information with available mailing lists against the rules set by the domain name administrators where:

1. Senders that maintain their own mailing lists, will use the subscribe server from time to time to update their mailing lists by removing anyone who has indicated that they do not wish to receive mail from the source, mail messages with a given classification.

   The mailing list update may be done in many ways, but for simplicity we assume that the sender will send a list of all the emails in the given domain name, along with the intended mail classification, the server will then process the list, and return a list of the ones that explicitly accepts receiving mail for this classification. These providers will then have to unsubscribe any users in their mailing list before attempting to email them.

2. Businesses that want to email marketing materials to users, can access the subscription server to get information on which accounts will be accepting mailing for a certain topic. The service of providing these email addresses could be free or paid.

3. The server will also be used to synchronize with other subscribe servers if they wish to syndicate or distribute their mailing list.

7. Delegating and authenticating Mail Holding Facilities

There are two ways to configure and setup an MHF [50] so it can be accepted by the recipientÆs Incoming Mail Gateways [70] (IMG) as a valid MHF for the domain it serves:

Static setup: used in simple setups.

The domain administrator has to explicitly delegate each MHF as a valid mail server by entering their name and IP in the DNS table in a specific format. The naming convention is mhf-N.domainname, where N is the any positive integer value. This convention is used to make it possible to identify servers that are acting as MHFs on behalf a domain.

When an MHF sends a notice to other AMDP servers, it identifies itself by entering its domain in AMDP-MHF-NAME header, which is authenticated by making a DNS lookup.

The ultimate goal is to eventually define a new MH record in the DNS specification that will explicitly delegate which servers can act as an MHF on behalf a domain.

Dynamic setup: used in complex setups involving dynamic, and long list of MHFs

The AMDP-MHF-NAME included in the envelope points to the domain name of the MIS [60]. The MIS holds the name and IP of all valid MHF for the domain.

The naming convention for the authentication MIS is amdp-mis.domainname. This will make it easier to visually know which servers act as MIS on behalf a domain.

The AMDP-MHF-ID is a character string unique to that domain, and maintained by the MIS server [60]. The flexibility of having the MIS autheticate the AMDP-MHF-ID is required in cases where redundant servers are in place and in the event a given MHF is down the administrators can setup alternate MHFs to assume the responsibility of the MSG-IDs that were under the care of the downed server.

This setup is much more flexible for administrators delegating hundreds or thousands of MHFs to third parties, and not having to wait for DNS to resolve.

In the final stage (optional):

In the long term, DNS should be upgraded to include: 1) An MH record type, which is equivalent to listing allowed MHFs in the DNS 2) An AX record that denotes the authentication servers to be used for this domain.

For example yahoo.com authorizes 10 servers to act as primary MHF

on its behalf by inserting them directly into its DNS table. It also maintains it own MHF authentication server (AUTH-MHF), which is also defined in their DNS table, and it handles another 500 MHF.

If any of the first 10 servers sends a message on behalf of Yahoo, it will include in the header an AMDP-MHF-NAME header pointing to its domain name (mhf-N.yahoo.com), letting the receiving AMDP know that it can make a direct query of the DNS and find its corresponding IP.

However if any of the other 500 servers managed by the AUTH-MHF sends any messages, then they submit their assigned number as AMDP-MHF-ID along with their authentications server domain name in the AMDP-MHF-NAME tag (example AMDP-MHF-ID: MHF-005 AMDP-MHF-NAME: auth-mhf-501.domainame). Once an MHF-ID is detected, then the IP of the MHF-NAME is queried from the DNS, and then it is contacted with MHF ID to obtain the domain IP of the MHF hosting the message..

8. Mail Classifications

Email classification is a method by which a sender pre-identifies the topic of an electronic message, so it can be properly processed for delivery through the various mail gateways it passes through. This classification is also used in the optimization of the delivery process and allocation of recourses based on the message priority, which is directly related to its classification. Classification is also a way to control the type of mail allowed to a given domain. For example a business can make known on their public policy that they will only receive mail from authenticated mail servers, and hence reduce the incoming mail to businesses and companies willing to be authenticated.

Classifications can be of two forms:
1. Authenticated classifications.
2. Non-Authenticated classifications.

When a message is classified using a properly authenticated classification service, the recipient mail server 70 honors the type of mail classification assigned. This can reduce the steps a receiving gateway will do in checking for abuse. This also shifts some of the burden to the authenticating third party, which will be involved in any disputes, and manage the active classification of the domain. It will also reclassify the domain based on its current activity as reported by receiving mail servers. However in non-authenticated classifications, more checks could be made, such as

querying other black lists

Structure of the classification string:

The mail classification string is composed of three or more parts that define the agent type, mail type, and message type. When put together, they provide a standard methodology to describe the message topic. The various types are separated by a common separator such as æ::Æ.

Agent type

Defines how the senderÆs mail server behaves. The mail server can be a public mail gateway and hence defined as "Public", the mail server could relay mail directly on behalf of its domain, and is named "Direct", while companies relying on third party servers to relay its mail, represent themselves as "Agent". As other new agent types surface, they can be assigned a key type.

This type is defined by the domain administrator but can be overwritten by third party services that authenticate the agent type for a given domain.

Mail type

The mail type defines the primary use of the domain, and is set by the mail administrator. For example a domain can be used for business, bulk, education, government, personal, etc.. This is an important classification, and initially will be allowed to be set by the owner of the domain, however it also allows the design to be adaptive and to be changed in the future to overwrite this by known activity of the domain name as kept by third party services such as todayÆs blacklists, which most likely has to occur since some domains will still feel the urge to forge their mail type classification. Some of the classifications are defined below:

Bulk - This means that this business mainly deals with bulk email, they are marketers, mass mailing sites, etc..

Business - This means the company is a business that deals with other business and it does not use this domain to mass mail, if they do occasionally send bulk mail, then they need to identify those messages as bulk. If the domain administrators, or its agent, fail to comply then it can be blocked from further mail processing.

Government - This means that the emails generated are mainly official in nature, and mass mailing is not expected from these.

Educational - This means that the emails generated are mainly

educational, and not to be confused with messages from .edu domains. A university may be classified as a business if its emails are to prospective students, while domains or MHFÆs used to server students email, can set this key to "Personal" or "Bulk" depending on the volume of messages generated.

Commerce - This means that the business is using this message to complete an e-commerce transaction, such as sending in a receipt, invoice, etc.. This domain name can not act as a bulk domain as it will loose its commerce classification.

Personal û This means that this domain is mainly used for personal purposes and does not Spam. If it does, it should use the Bulk key instead.

Message type

This is the sub category of mail types and it is defined by the user. It lets the final recipient know what type of message it is, and it used for informational purposes only, since there is no way to guarantee that the field is used correctly. However it is included for completeness, making it possible for businesses to use this flag for more advanced functionalities in email. Some of possible message types are included here for illustration purposes:

BULK: Legal, Adult, Entertainment, Mailing List, Travel, Computers, Sports
BUSINESS: Inquiry, Response, Proposal, Personal
GOVERMNET: Subpoena, Information, Request
COMMERCE: Invoice, Receipt

Recommendations:

It is recommended that a domain name use a separate domain name for bulk mailing where they plan to mail hundreds or more unsolicited or solicited messages. For example yahoo.com would use yahoo-bulk.com when emailing few million users.

Mail Classification Examples

In the following examples the ADMP envelope header will look something like these:

MAIL-CLASS: Public::Bulk::Entertainment (1) or,
MAIL-CLASS: Direct::Business::Internet Service::Inquiry (2) or,
MAIL-CLASS: Agent::Bulk::Mailing List (3)

In these examples, the separator used is the æ::Æ. Example (1) is originating from a "Public" mail gateway such as Yahoo.com, or Ayna.com. The message is classified as "Bulk" by the administrators of the mail system, since they do not have control of publicly available email accounts. The user has announced that this message is of a "Entertainment" topic, and has been accepted as such.

Although in example (1) we said that Public gateways send Bulk mail that was just an example, since Yahoo can decide to make available a paid account in which the user tells the mail provider, in this case Yahoo that this account is to be used for business or personal.

In example (2) a domain name used and managed by a company, sets its agent type to "Direct" since they process their own mail, while the domain is a "Business" domain, and it classifies itself as "Internet Service", and the user mailing

9. Automatic reporting and resolution of classification abuse

The proposed system to be used allows mail servers to transmit abuse reports which can generate notifications send to the administrator of the domain, and a way to resolve those issues.

Most spammers get away with what they do, because there is no one way to find out who did what. A spammer who spams one or more domains will trigger automatic notifications to the reporting gateway, which will use a specific criteria to issue an alert or a warning to other mail systems not to receive mail generating from this domain, or to redefine parts of the mail classification. This entity can become a legal traffic entity dealing with issues arising from mail abuse, and can be supervised by the US postal service as it is done today.

10. AMDP envelope headers

The protocol used by AMDP follows the same guidelines as SMTP. Most of SMTP commands are used, while others were added or used in a different way.

AMDP relies on sending routing and authentication information using a standard format referred to as an AMDP envelope. It is similar to one used in SMTP, however some changes were made to keep AMDP flexible and be able to grow without the restrictions imposed by certain SMTP headers.

10.1 AMDP-About

This header is used to provide a URL to the website who develops and maintain the AMDP implementation used to mail the mail message.

Usage

AMDP-About: http://www.foo.com/

10.2 AMDP-From, AMDP-From-Name

The AMDP-FROM contains the email address of the user sending the message, while AMDP-FROM NAME contains the name of sender.

In SMTP envelopes, both the email and name of sender are transmitted under the same header (FROM). Using one heading for the two pieces of information, adds several complications when it comes to encoding and decoding the name or address for internationalization purposes.

In AMDP these two pieces of data are split into two fields the AMDP-FROM and AMDP-FROM-NAME.

The email address AMDP-FROM of the sender is examined and corrected by the sender's Outgoing Mail Gateway [20] after authentication. This reduces common problems due to the inexperience of some users of the proper notation of their email, especially when we enter the realm of international domain names, and possibly dynamically assigned email addresses used for secure transactions, or to provide anonymous email services.
It is also a mechanism to overwrite any attempt made by internal networks to forge their identity in outgoing mail messages.

The AMDP-FROM NAME is not set by OMG [20] since the user can change the name on the account without affecting the functionality. This field is optional.

Usage

AMDP-FROM: <adonis@amdpmail.com>

The header contains the email address enclosed by <> and encoded in UTF8.

AMDP-FROM NAME: <Last name, First Name>

The header contains the name of the user in UTF8 enclosed by <>.

10.3 AMDP-To, AMDP-To-Name

The AMDP-TO header contains the recipient's email address, while AMDP-TO-NAME contains the recipient name.

In SMTP envelopes, both email and name of recipient are transmitted together under the same header (TO). Using one heading for the two pieces of information adds many complications when it comes to the encoding of the name or address for internationalizations purposes.

In AMDP these two pieces of data are split into two fields the AMDP-TO and AMDP-TO-NAME.

The email address of the recipient is entered by the user, and is converted to UTF8 by the email client [10] or the mail gateway [20].

Usage

AMDP-TO: <email@address.com>

The header will contain the email address between <> and it is in UTF8. Which will enable older systems that support ASCII based email addresses to pass through it easily, while allowing new mail systems that are UTF8 enabled to process the new names.

AMDP-TO NAME: <Last name, First Name>

This command will contain the names of the users. It is in UTF8. Since each envelope sent out from MHF [50] belongs to one and only one email user, AMDP-TO and AMDP-TO-NAME do not include usage examples of multiple recipients, however when an email is being sent from the OMG [20] to the MHF [50], multiple recipients are acceptable and the usage is defines as such:

AMDP-TO: <email1><email2><email3>
AMDP-TO-NAME: <name1><><name3>

Where the name of email2 was not specified

10.4 AMDP-SUBJECT

AMDP-SUBJECT: Email subject

The subject specified the subject of the message and should be converted to UTF8. AMDP does not use the SUBJECT set by SMTP since it may contain various encodings that will limit the functionality of

the subject field. In the absence of AMDP-SUBJECT, the implementation can use SUBJECT provided by SMTP for that purposes.

Usage

AMDP-SUBJECT: Subject Title of the message

10.5 AMDP-ATTACHMENTS

ATTACHMENT: Email attachment list

This is the list of attachments used in a message. (TBD)

Usage

AMDP-ATTACHMENT:

10.6 AMDP-Mail-Class

This header is used to classify the types of emails sent. Please refer to section . 8 titled "Mail Classifications" for details about the mail classification.

The mail classification string is composed of three or more parts that define the agent type, mail type, and message type. When put together, they provide a standard methodology to describe the message topic

The first part is mandatory and is usually controlled by the service provider, while the second part is controlled by the sender. So an ISP can classify accounts as business or personal, and the user can set another level of classification if they want to.

Usage

AMDP-MAIL-CLASS: Public::Bulk::Entertainment

Security considerations

The classification is prone to being abused and need to have various checks. The design assumes that third party services will be created to acknowledge if a given domain falls within an agent type or another. It will also promote or demote the mail classification of a given domain.

10.7 AMDP-Language

This header defines the language of the content of the email. It is set to the ISO 639 code.

It can be set by the email client [20], or by the MHF [50] that can use the UTF8 ranges to determine the language of the message, or use more complex language pattern matching algorithms.

This is needed to match the server messages to the language of the mail message in case of an error, and also to automate the process of translation from one language to another in the future.

Usage

```
AMDP-LANGUAGE: EN
```

10.8 AMDP-Encoding

The ENCODING header will be set to the industry standards encoding codes as defined by Unicode documentation, and RFC 2277 iv

It is used to identify the encoding of the mail message. Ideally this should be set to UTF8, but it is available to be set to whatever the system supports and allows for other programs to know how to deal with the encoding of the message.

Ideally the MHF should convert all messages it receives into a UTF8 encoding message unless it is encrypted by the original sender. This will make it easy for the mail receiver efforts to focus on Unicode support.

Usage

```
AMDP-ENCODING: UTF-8
```

10.9 AMDP-Date

This is the data the message was sent from the user [10] in UTC format.

Usage

```
AMDP-DATE: 2003-01-29 10:10:10 AM UTC
```

10.10 AMDP-OMG-ID

This is the unique identification string for the outgoing mail gateway OMG [20].

It is mainly used in communications between OMG [20] and MHF [50], and it is not forwarded outside the domain. It is also used to route messages to the sender.

Usage

```
AMDP-OMG-ID: foo.bar.com
```

10.11 AMDP-MSG-Id

This is a unique id issued by the MHF [50] holding the message. The number is one of the keys used to retrieve the message by the intended recipient. In the event a sender sends one message and copies five people. Each recipient will receive an envelope with a different MSG-ID. This id is an important tool is reducing Spam and the manner it is created should by chosen carefully as not to be predictable.

Note: The MSG-ID is used once and can not be reused until it has expired, it MUST be difficult to guess, and SHOULD NOT be built upon any relation with the other pieces of information used to authenticate i.e. email address, expiration, and timestamp. This will make it much more difficult to impersonate others online.

Usage

```
AMDP-MSG-ID: MMXA-12348754-ABVGS
```

10.12 AMDP-Size

This is the size of the mail message, including the attachment. It should include the units used.

Usage

```
AMDP-SIZE: 4068 <MB>
```

10.13 AMDP-MHF-Name, AMDP-MHF-Id

The two vales are used to identify the Mail Holding facility [50] to the recipient servers [70]. Please refer to section 7 for a detail description of these fields and the values to be assigned to them.

10.14 AMDP-Auth-Port

The header contains the port number at which the authentication server defined in AMDP-MHF-Name will return queries about messages received from a specific MHF [50]

10.15 AMDP-Timestamp

The header contains the timestamp at which the envelope was created by MHF, before contacting the IMG [70]. It is used as part of the authentication scheme.

10.16 AMDP-Expire-On

The header contains the date the message expires on. This will allow both sides of the delivery process to clear expired messages.

10.17 AMDP-Auth-Keys

This is the list of authentication keys that need to be used by the IMG [70] when contacting the MHF [50]. If not specified then the key are assumed to be AMDP-TEMPSTAMP, AMDP-EXPIRE-ON, and AMDP-MSG-ID

10.18 AMDP-VERSION

This includes the version number of AMDP used in the envelope. This allows servers to negotiate advanced features as they become available, and be adaptable to earlier versions of the protocol

10.19 AMDP-PAYMENT-RCPT

This will inform receiving servers that postage has been made, and provide the information needed to retrieve the payment information.

Security Considerations

Security considerations are outlined in the AMDP design section within the appropriate context.

References

i Bradner, S., "The Internet Standards Process", BCP 9, RFC 2026, October 1996.

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

iii Postel, J., "Simple Mail Transfer Protocol", RFC 821, August 1982.

iv Alvestrand, H. "IETF Policy on Character Sets and Languages ", BCP 18, RFC 2277, January 1998

Acknowledgments

<Add any acknowledgements>

Author's Addresses

Adonis El Fakih
PO BOX 6048
NASHUA, NH 03063, USA
phone: +1 (508) 801-0273
Email: adonis@aynacorp.com

IPR Notices

Intellectual Property Rights disclosure statement pertaining to AMDP

Adonis El Fakih has a patent pending that may relate to AMDP internet draft specifically to the work derived from draft-amdp-00.txt.

Upon approval by the IESG of the relevant Internet standards track specification and if any patents issue to A. El Fakih with claims that are necessary for practicing this standard, any party will be able to obtain the right to implement, use and distribute the technology or works when implementing, using or distributing technology based upon the Specific specification(s) under openly specified, reasonable, non-discriminatory terms.

Because A. El Fakih wants to make this mail delivery protocol widely available to help control the growing problems associated with unsolicited mail, the non-commercial use of this protocol is free.

Requests may be sent to:
Adonis El Fakih
PO BOX 6048
Nashua, NH 03063, USA
phone: +1 (508) 801 0273
email: adonis@aynacorp.com

Copyright Notice