Internet Draft
Document: draft-crouzet-amtp-01.txt
Expires: April 2004

B. Crouzet
Institute of Technology
Tallaght

October 2003

Authenticated Mail Transfer Protocol

Status of this Memo

Abstract

Recent years have seen electronic mail becomes the first mode of communication. Electronic mail allows users to exchange information using different formats such as files, pictures, videos and text messages. Electronic mail is quicker and faster than other modes of communication but it generates more problems, such as email bombing, email virus, email spoofing, anonymous email, relaying email and in particular spam email.

This Internet Draft aims at solving or reducing the above problems by proposing a new transfer protocol, Authenticated Mail Transfer Protocol. Authenticated Mail Transfer Protocol is a second modified version of the current transfer protocol, Simple Mail Transfer Protocol. It identifies a sender, differentiates a server from a user, changes the electronic mail structure and improves the electronic mail transaction.

The purpose of this document is to describe Authenticated Mail Transfer Protocol to the Internet community. The implementation of the new transfer protocol allows us to carry out a series of tests that measures and proves the performance and efficiency of the new transfer protocol.

Conventions used in this document

SA => Sender Server: SA represents an email server where the sender is known.

SB => Recipient Server: SB represents an email server where the recipient is located.

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

Table of Contents

1. Introduction

The last solution is to add an identification state to the Simple Mail Transfer Protocol, creating a new protocol to be called Authenticated Mail Transfer Protocol (AMTP). It would use the Transmission Control Protocol/Internet Protocol (TCP/IP) model to communicate across the network. AMTP is a five state process with three Client-to-Server communication states (Identified, Email and Logout states) and two Server-to-Server communication states (Information and Retrieved states).

The first Client-to-Server state is the Identified state where the protocol asks for a username and a password to identify the user. The user has to log in successfully before he/she can use the server. The second state is the Email state where the user can employ any protocol commands in order to send an email. Once the user has logged onto the server, he/she does not have to enter his/her email address any more. The server will automatically add his/her email address to the email. The last state is the Logout state where the user is logged onto the system therefore he/she has to be logged out.

There is also a new transaction between two email servers. The first Server-to-Server state is the Information state, whereby the sender server informs the recipient server that an email is waiting to be retrieved. The second state is the Retrieved state, whereby the recipient server retrieves the email from the sender server.

Authenticated Mail Transfer Protocol will structure the email in order to provide an easier way to find information inside the email, to limit the access of the email by a user and finally, to validate the email itself. The structure of the email will allow a filter to be more efficient in order to look the most important fields. Authenticated Mail Transfer Protocol adds three XML tags that separate the information inside the email.

A relaying server allows a sender server to route an email without knowing the address of the recipient server. A relaying server transmits the email to the recipient server or another relaying server like a normal Server-to-Server communication. In order to protect a network, it is possible to use a router, a firewall or a proxy server associated with a firewall. Under these protections, AMTP is operational. AMTP does not work behind a proxy server.

Authenticated Mail Transfer Protocol both adds and removes commands from SMTP. The Authenticated Mail Transfer Protocol procedures are

demonstrated. It also adds new reply codes and uses identical reply codes from SMTP.

This document explains the result of the series of tests that Authenticated Mail Transfer Protocol passed. Authenticated Mail Transfer Protocol is compared to Simple Mail Transfer Protocol in order to prove that the new protocol solves three email problems: anonymous, spoofed and relaying email. Four network simulations, a speed evaluation of these two protocols and different attacks against the new protocol have been carried out and tested. This demonstrates the efficiency and the performance of the new transfer protocol.

In the Appendix chapter, acronyms and terminology are defined in Appendix A and B.

2. Presentation of Authenticated Mail Transfer Protocol (AMTP)

Authenticated Mail Transfer Protocol (AMTP) is located at the Application layer of the Transmission Control Protocol/Internet Protocol (TCP/IP). The AMTP header handles and presents the data to the user. It analyses commands and replies to the user.

2.1 General View of Authenticated Mail Transfer Protocol

The following figure gives a general view of the Authenticated Mail Transfer Protocol states. In the Identified state, the user has to be identified before he/she sends the email. The user writes his/her email in the Email state. At the end of the email, the server delivers the email to the recipient. If the recipient is internal, the email is immediately delivered to the recipient mailbox. If the recipient is external, the sender server uses the Information state. The recipient server executes the Retrieved state in order to retrieve the email from the sender server. These two states are reserved for an email server and are the result of the solution to identify a sender and validate a sender email address.

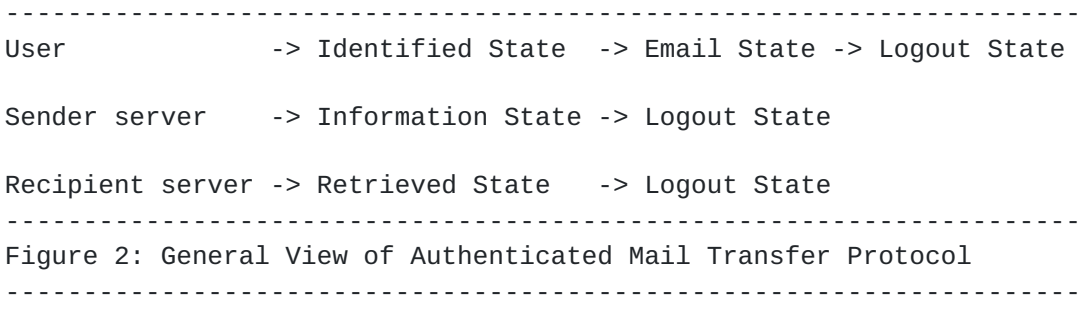

Figure 2: General View of Authenticated Mail Transfer Protocol

2.2 Explanation of Authenticated Mail Transfer Protocol

The host server starts the Authenticated Mail Transfer Protocol service by listening to port 26. The email client establishes a TCP connection with the AMTP server. If the AMTP server accepts the

connection, it sends back a reply code 220. Now the AMTP server and the email client can exchange commands and replies. The AMTP server or the email client with the QUIT command can close or abort the transaction at any time.

The Authenticated Mail Transfer Protocol session progresses through a number of steps during its lifetime. Once the TCP connection has been opened and the AMTP server has accepted the connection, the session enters into the Identified state. In this state, the user must identify himself/herself to the AMTP server. Once the user has successfully done this, the session enters into the Email state. In this state, the user will be allowed to request an action from the AMTP server. He/she can send an email to a random user. When the user has finished his/her session, he/she has to enter the QUIT command and the session enters into the Logout state, i.e. the AMTP server closes resources used such as the network connection (TCP), the database connection and the files.

Authenticated Mail Transfer Protocol contains two server states: Information and Retrieved states. These states are reserved for the AMTP server only and occur in cases where the AMTP server has to send an external email. A user will be able to use the command but the transaction will be aborted after the AMTP server recognises the parameters are incorrect. Only one AMTP server, the sender server, recognises the user. The recipient server accepts information coming from the sender server or a user. The only thing email servers have in common is port 25. It is the only piece of information that an email server can recognise from another email server. Port 25 makes sure that the transaction takes place between two email servers and not between a user and a server.

The first state is the Information state. In this state, the sender server informs the recipient server that an email is waiting to be delivered. The sender server gives the recipient server a number that refers to the email, the recipient email address and its IP address or domain name. The IP address or domain name is required to allow the recipient server to connect on to the sender server. The second state is the Retrieved state. In this state, the recipient server connects to the sender server to retrieve an email. It gives the number and the recipient email address that has been transmitted in the Information state. When these states are completed, the email servers close all resources used.

These two states are automatic and fast because it is simply two computers that exchange data. Time out can be created when the server is waiting for a command. The two functions of these email servers are to send and read information and as such, they do not perform tasks that require time, resources or memory.

2.3 Goals

This solution solves the problem of anonymous and spoofed email, and reduces the effect of email virus, email bombing and spam email. Therefore, everyone knows where the email comes from. The sender exists and the sender server recognises him/her. It does not stop spam email but a user has the possibility to avoid it and locate the sender. The solution still has to be tested to see if a hacker can crack it and if this solution is feasible on the network.

3. Authenticated Mail Transfer Protocol States

3.1 Identified State

3.1.1 Presentation of the Identified State

This state is important because it protects a user from spoofed and anonymous email. Two new commands have been added to implement this state: USER and <USERNAME> <PASSWORD>. When the AMTP server accepts the connection, the user enters into the Identified state.

In this state, the user types the USER command that means he/she wants to be identified by the AMTP server. The AMTP server answers with the reply code 250 when it is ready to identify the user. The user types his/her username and password as a simple command. Any user can be identified with these parameters. There are three types of answers for the server. In the case of the username and password corresponding to one user, the server replies with the code 250 and sends him/her helpful server information. The AMTP server now identifies the user who can now use any AMTP commands. In the case of the username and password being incorrect, the server replies with the code 401. After the third try from the user, the server closes the connection and replies with the code 555.

3.1.2 Command in the Identified State

USER

The user enters the USER command to inform the AMTP server that he/she wants to be identified by the server. The USER command does not need any parameters.

<USERNAME> <PASSWORD>

The user needs to enter his/her username and password, which contain no space in them. In order to protect the user, the username has to be different from his/her email address. In addition to this, the username and password are entered after the USER command in order to protect this data. It will be difficult for a hacker to find these parameters. If the hacker is watching the network, he/she has to catch the USER command and the packet with the username and password data, which contains no sign of them.

3.2 Email State

3.2.1 Presentation of the Email State

In this state, the user can send an email. Once the user is logged into the AMTP server, he/she does not have to enter his/her email address any more. The AMTP server adds his/her email address into the email header. The MAIL FROM command has been removed from the transfer protocol. The RCPT TO and DATA command from SMTP are still used in this state. Two new commands, MORE TO <Recipient email address> and HEAD, have been added in order to improve the service of the transfer protocol.

The user enters a recipient email address using the command: RCPT TO: <Recipient email address>. The AMTP server validates the email address and acknowledges if the recipient email address is internal or external to the system. If it is internal, the server checks if the user exists or not and sends back the appropriate reply code: an error message in case the user does not belong to the server. If it is external to the system, the AMTP server does not check the recipient email address, it just validates it. The AMTP server continues the process and the user enters the DATA command to specify the email body. The user writes the content and the AMTP server stores his/her data on an email. If the recipient email address is internal, the AMTP server transports directly the email to the recipient mailbox. If the recipient email address is external, the AMTP server starts the Information state.

3.2.2 Command in the Email State

RCPT TO: <recipient email address> [, <recipient email address>]
This RCPT TO command is used to identify an individual recipient of the email. It is the same command described in RFC 2821 [2], therefore reply codes are the same. The parameter for this command can be a list of recipient email addresses separated by a comma (æ,Æ). The command returns information about the validity of the recipient email address.

DATA
The user uses this command to enter the email content. When the AMTP server accepts the DATA command, it has to send an email to the recipient. It is the same command described in RFC 2821 [2]. Reply codes are the same.

3.3 Logout State

3.3.1 Presentation of the Logout State

The Logout state is used to close the connection between the AMTP server and the email client when the user has finished with his/her email and wants to leave the AMTP server. The AMTP server closes all resources used like the database, the TCP channel, the files and the thread. The user uses the QUIT command to terminate the transaction.

3.3.2 Command in the Logout State

QUIT

The QUIT command does not need any parameter. The server replies with the code 221. It is only after this reply code that the transaction is finished. It is the same command described in RFC 2821 [2], therefore reply codes are the same.

3.4 Information State

3.4.1 Presentation of the Information State

In this state reserved for the server only, the sender server contacts the recipient server. The sender server connects to the recipient server and receives the reply code 220. Then, the sender server sends the SELO command with three parameters that are the domain name of the sender server, a unique number created by the sender server and the recipient email address. The recipient server verifies if the domain name or IP address corresponds to the parameters found in the network packet. It checks if the recipient email address exists or not in its server. If the recipient email address is unknown, the recipient server sends the error back to the sender server; the sender server sends it back to the user and deletes the email. If the process is handled successfully, the recipient server continues with the Retrieved state.

In the case of the server being a relay to distribute the email, the sender server proceeds normally. The relaying server will retrieve the email and send the number to the recipient server. The sender server will only proceed with the relaying server. The relaying server does not need to check the recipient email address. In a relaying server, the domain name is the only barrier that could stop an email from being sent.

3.4.2 Command in the Information State

SELO <DOMAIN> <NUMBER> <RECIPIENT EMAIL ADDRESS>

The SELO command needs three parameters: DOMAIN, NUMBER and RECIPIENT EMAIL ADDRESS. The DOMAIN parameter can be the IP address or the domain name of the sender server. It allows the recipient server to establish a connection with the sender server. The NUMBER parameter is the email identifier. This parameter allows the sender server to recognise the email. The RECIPIENT EMAIL ADDRESS parameter is used to identify a recipient email address. The sender server saves these parameters and the address of the recipient server. If the recipient server knows the recipient email address or if the domain name is in the relaying table, it sends back answers by the reply code 250. If these two conditions are not met, the recipient server answers with the reply code 550. If it is successful, the recipient server enters into the Retrieved state.

3.5 Retrieved State

3.5.1 Presentation of the Retrieved State

In the Retrieved state, the recipient server establishes a connection with the sender server to retrieve the email with the number given in the Information state. If the connection is successful, the sender server answers by the reply code 220. Then, the recipient server sends the SEMA command with two parameters separated by a colon (æ:Æ). These two parameters are the recipient email address and the unique number. The sender server checks if these parameters exist or not in its email queue. If the number, the address of the recipient server and recipient email address are correct, the message will be given to the recipient server. The recipient server stores the email and the email appears in the recipient mailbox. In case the number and the recipient email address are incorrect, the sender server sends an error message to the recipient server.

The relaying server proceeds through this state. The difference between a relaying server and a recipient server is that the relaying server will start the Information State to inform another relaying server or the recipient server. The relaying server will store the email and create a number to use the SELO command. It implements a relay queue to keep sending the email.

3.5.2 Command in the Retrieved State

SEMA <RECIPIENT EMAIL ADDRESS >:<NUMBER>

This command is reserved for the AMTP server. It needs two parameters: RECIPIENT EMAIL ADDRESS and NUMBER. The RECIPIENT EMAIL ADDRESS parameter is the recipient email address. The NUMBER parameter is a number used to recognise the email on the sender server. If the number exists, the sender server will send the data together. If the number is wrong, the connection will be closed.

If the email has not been retrieved and the lifetime of the email has expired, the AMTP server will inform the sender about it. The sender can resend the email. The AMTP server will keep evidence of this email and inform the administrator about the fact that the email has not been retrieved. The administrator can think about the reason why the email was not delivered.

4. Structure of an email in the Authenticated Mail Transfer Protocol header

The email client needs to make a distinction between the email header, the relaying information and the email and its body. When an email client writes an email, there is a small distinction between the email header information and the email body. For example, the subject is entered with the email body. This option is technical and a user will not see the difference in the email client.

Authenticated Mail Transfer Protocol modifies the structure of the email. The header will be entered separately from the data.

AMTP adds the version of the transfer protocol used into the server information: Version 1.0 for SMTP and Version 2.0 for AMTP. By adding this parameter, a recipient server can prevent a user from risks incurred with SMTP. An AMTP server can accept an email from a SMTP server and insert the version of the protocol into the server information. The server information has the same content as the header field ôReceived Fromö in SMTP.

Using XML tags in the email, the server will be able to directly detect the information it needs. The sender server enters these tags. These XML tags are:
=> <RELAY> contains the relaying server information </RELAY>.
=> <HEAD> contains the header information of the email </HEAD>.
=> <BODY> contains the body information of the email </BODY>.

4.1 Relay Tag

In the relay tag, the information about the relaying server is specified. The relaying server should enter information about the sender server and the recipient server using the header field ôRELAY FROM: <sender server information> TO <recipient server information>ö. The recipient server information or the sender server information could be a relaying server. With this information, it will be possible to identify a relaying server from the sender server and to determine the route of the email.

4.2 Head Tag

In the head tag, the information about the email is specified. A new header field is introduced in order to distinguish the sender server. The line ôSend From:ö is used to display the sender server information. To avoid a hacker entering data in the email header, this head tag is reserved for the sender server. To implement this solution, an order of lines will be specified. This order will ensure the email is correct.

The order is:
=> Sender details: The line ôFrom: sender email address < name >ö.
=> Sender server: The line ôSend From:ö with the server information and the version of the transfer protocol used.
=> Date: When the email was written.
=> Message Identifier: The line ôMessage id: <number>ö.
=> Recipient details: The line ôTo: recipient email address < name >ö.
=> Subject: It is the subject of the email.
=> Other header: These lines are used to enter different headers that are not necessary for the delivery of an email.

=> MIME details: The details for the MIME protocol.

In order to distinguish the email header information from the email body information, the HEAD command is introduced. The user uses this command to enter MIME type information. For a simple text message in ASCII characters, the user can enter the header ôsubjectö into the email content using the DATA command. The header ôsubjectö will be added into the head tag. If a user does not type the HEAD command, the server detects a simple email and presents the email header correctly. The server adds the line ôsubjectö into the email and the content will be entered into the body tag.

If a user enters the HEAD command, he can type his/her information about the email. The first lines of the header are reserved for the server. The server adds the header: ôFromö, ôSend Fromö, ôDateö, ôMessage idö, and ôToö. After these lines, the user inserts header data that can contain different information for instance a MIME type.

4.3 Body Tag

The body tag is used to enter the email content. Any information in this tag will be considered as body information. This information will be displayed to the recipient as the email part.

4.4 Example of an email structure

The format of the email is:

```
<RELAY>
Relay from: < master.com (193.1.124.54); 06 June 2003 15:55:12
o'clock IST; Version: 2.0> TO <master.org (200.200.200.100)>
</RELAY>
<HEAD>
From: bcrouzet@master.com
Send From: master.com (193.1.124.54); 06 June 2003 15:55:12 o'clock
IST; Version: 2.0
Date: 06 June 2003 15:55:12 o'clock IST
Message id: 1055034769259
To: 2@master.org
Subject: XML Tags
</HEAD>
<BODY>
It is a demonstration of the new mail structure with three XML tags.
</BODY>
```

The advantage of these XML tags is to preserve the structure of the email. A user cannot change the information during the process of transferring an email. The sender server completes the XML tags HEAD and BODY. The recipient server and the relaying server modify the XML tag RELAY. Useful information is classified and available without searching in the email. AMTP adds an email structure in

order to detect the relay, header and body information quicker that Simple Mail Transfer Protocol.

5. Authenticated Mail Transfer Protocol Commands

5.1 Optional Commands

RSET

The RSET command allows a user to reset any action that has already been activated. It allows a user to restart the transaction from the beginning. It is the same command described in RFC 2821 [2]. The reply codes are identical.

NOOP

The NOOP command allows a user to reset the timer. It is the same command described in RFC 2821 [2]. The reply codes are the same.

HELP [<topic>]

The HELP command gives a user some information about the command it provides. It provides useful information for the client. It is the same command described in RFC 2821 [2]. The reply codes are identical. If a user enters a topic as a parameter, the system provides information on this topic.

MORE TO: <recipient email addresses> [, <recipient email address>]

The MORE TO command allows a sender to add more recipient email addresses to the email without changing the first recipient email address entered or to correct a recipient email address entered wrongly with the RCPT TO command. The RCPT TO command does not correct invalid email address. The MORE TO command can be used after the RCPT TO command and does not replace the RCPT TO command. Using the MORE TO command, the sender corrects invalid email addresses. The parameter <recipient email addresses> specifies multiple recipient email addresses separated by a comma (æ,Æ). This command gives the validity of the recipient email address.

The advantage of the MORE TO command is to prevent a user from retyping a valid email address using the RCPT TO command. A user can enter the MORE TO command to add different recipients to the email. The programme has to store all valid recipient email addresses during the process. This could be a little inconvenient but this command saves time in the overall process.

HEAD

A user types the HEAD command to enter the email header details. This command is like the DATA command, is entered before the DATA command and separates the email header from its body. The server replies with the code 354 to enter the email header details. To finish entering the email header details, the user enters a dot (æ.Æ). The AMTP server stores the email header and wait for the DATA

command. The email header is always in American Standard Code for

Information Interchange (ASCII) characters and no code has to be given in reply to the user.

The advantage of the HEAD command is to differentiate between the header information to the body information. This command specifies the header information of the email. The user or the programme uses it when the email is complex. The drawback is that the programme or the user has to enter the header information into this command during the Client-to-Server transaction; it therefore adds time to the overall process.

5.2 Obsolete Commands

MAIL FROM

The sender server manages this command and adds the sender email address to the email. It is a hidden field like the ôreceived fromö field.

EHLO

Since a user has to be identified by the server, there is no point to keep this command but the result of the EHLO command is important. It gives helpful information about the server to the user. The result will be displayed after the user has been identified.

TURN

This command allows a client to become a server and the server to become the client. For security reasons, this command has been disabled.

VRFY

A user will be unable to verify an email address for security reasons. It is important to know and check an email address but today phone, letter or email communications can transmit email addresses.

EXPN

For security reasons, this command has been removed from the protocol. This command confirms that the argument is a mailing list. It is dangerous because a user can know the name of a mailing list and diffuse it.

HELO

This command comes from RFC 821 [1] and been replaced in RFC 2821 [2] by the EHLO command. There is no point in keeping this command in the protocol.

SEND

It is rarely implemented. There is no point in keeping this command and since the protocol changed, this command is obsolete.

SOML

It is rarely implemented. There is no point in keeping this command and since the protocol changed, this command is obsolete.

SAML
It is rarely implemented. There is no point in keeping this command and since the protocol changed, this command is obsolete.

5.3 Order of commands

There are restrictions on the order in which these commands may be used. A session starts with the USER command. After this, a user enters his/her username and password. The server accepts the client if he/she is identified and lets him/her continue the transaction. The NOOP, HELP and RSET commands can be used at any time during a session or without previously initialising a session.

The RCPT TO command begins the construction of the email. It specifies the recipient email address or multiple recipient email addresses. A user can add more email addresses with the MORE TO command that also allows a user to correct an email address. If a user has a complex email header, he/she enters the HEAD command. He/she continues in any case with the DATA command to send the email. The transaction can be aborted by the RSET command. There may be zero or more email in the session.

To close the connection, a user types the QUIT command. He/she requests the end of the session.

5.4 Authenticated Mail Transfer Protocol Procedures

5.4.1 Simple Procedure

A simple AMTP procedure for a user is:
S: 220 AMTP >> Connection successful.
S: 250 AMTP >> Received from: postgrad-bc 193.1.124.54.
S: 250 AMTP >>
C: user
S: 250 AMTP >> Server Ready
C: bct 123
S: 250 AMTP >> Welcome Brice CROUZET to the AMTP server.
S: 250 AMTP >> SERVER INFORMATION.
S: 250 AMTP >>
C: rcpt to:jdoody@master.com
S: 250 Recipient accepted for "jdoody@master.com"
To add or correct a recipient address, please use the command MORE TO
S: 250 AMTP >>
C: data
S: 354 Enter the data of the message. End with "." on a line by itself.

```
C: Subject: AMTP Procedure 1
C: It is a simple AMTP procedure.
C: .
S: 250 Mail delivery successful for "jdoody@master.com"
S: 250 AMTP >>
C: quit
S: 221 Disconnection

The email has been received:
<RELAY>
</RELAY>
<HEAD>
From: bcrouzet@master.com
Send From: master.com (193.1.124.54); 09 April 2003 08:56:50 o'clock
IST; Version: 2.0
Date: 09 April 2003 08:56:50 o'clock IST
Message id: 1049998467218
To: jdoody@master.com
Subject: AMTP Procedure 1
</HEAD>
<BODY>
It is a simple AMTP procedure.
</BODY>
```

5.4.2 Procedure using optional commands

An AMTP procedure using optional commands is:

```
S: 220 AMTP >> Connection successful.
S: 250 AMTP >> Received from: postgrad-bc 193.1.124.54.
S: 250 AMTP >>
C: user
S: 250 AMTP >> Server Ready
C: bct 123
S: 250 AMTP >> Welcome Brice CROUZET to the AMTP server.
S: 250 AMTP >> SERVER INFORMATION.
S: 250 AMTP >>
C: help
S: 214 This is an AMTP Server.
214 Topics:
214 QUIT        HELP        RCPT        HEAD        DATA        RSET
NOOP
S: 250 AMTP >>
C: help data
S: help for DATA
S: Explanation
S: 250 AMTP >>
C: noop
S: 250 AMTP >> Noop OK
S: 250 AMTP >>
```

```
   C: rcpt to:jdoody@master.com
```

```
S: 250 Recipient accepted for "jdoody@master.com"
To add or correct a recipient address, please use the command MORE
TO
S: 250 AMTP >>
C: more to:bcrouzet@master.com
S: 250 Recipient accepted for "bcrouzet@master.com"
To add or correct a recipient address, please use the command MORE
TO
S: 250 AMTP >>
C: head
S: 354 Enter the header of the message. End with "." on a line by
itself.
C: Subject: AMTP Procedure 2
C: .
S: 250 Head Command Accepted
S: 250 AMTP >>
C: data
S: 354 Enter the data of the message. End with "." on a line by
itself.
C: Subject: Test
C: It is an AMTP procedure using optional commands.
C: .
S: 250 Mail delivery successful for "jdoody@master.com",
"bcrouzet@master.com"
S: 250 AMTP >>
C: quit
S: 221 Disconnection

The email for jdoody@master.com has been received:
<RELAY>
</RELAY>
<HEAD>
From: bcrouzet@master.com
Send From: master.com (193.1.124.54); 09 April 2003 09:00:17 o'clock
IST; Version: 2.0
Date: 09 April 2003 09:00:17 o'clock IST
Message id: 1049998673855
To: jdoody@master.com
Subject: AMTP Procedure 2
</HEAD>
<BODY>
Subject: Test
It is an AMTP procedure using optional commands.
</BODY>

The email for bcrouzet@master.com has been received:
<RELAY>
</RELAY>
```

<HEAD>

```
From: bcrouzet@master.com
Send From: master.com (193.1.124.54); 09 April 2003 09:00:17 o'clock
IST; Version: 2.0
Date: 09 April 2003 09:00:17 o'clock IST
Message id: 1049998673895
To: bcrouzet@master.com
Subject: AMTP Procedure 2
</HEAD>
<BODY>
Subject: Test
It is an AMTP procedure using optional commands.
</BODY>
```

5.4.3 Procedure with RSET command

```
An AMTP procedure using the RSET command is:
S: 220 AMTP >> Connection successful.
S: 250 AMTP >> Received from: postgrad-bc 193.1.124.54.
S: 250 AMTP >>
C: user
S: 250 AMTP >> Server Ready
C: bct 123
S: 250 AMTP >> Welcome Brice CROUZET to the AMTP server.
S: 250 AMTP >> SERVER INFORMATION.
S: 250 AMTP >>
C: rcpt to:jdoody@master.com
S: 250 Recipient accepted for "jdoody@master.com"
To add or correct a recipient address, please use the command MORE
TO
S: 250 AMTP >>
C: rset
S: 250 AMTP >> Reset OK
S: 250 AMTP >>
C: data
S: 503 Need RCPT before DATA "data".
S: 250 AMTP >>
C: rcpt to:2@master.org
S: 250 Recipient accepted for "2@master.org"
To add or correct a recipient address, please use the command MORE
TO
S: 250 AMTP >>
C: data
S: 354 Enter the data of the message. End with "." on a line by
itself.
C: Subject: AMTP Procedure 3
C: It is an AMTP procedure using RSET command.
C: .
S: 250 Mail in the spool for delivery for "2@master.org".
```

```
   S: 250 AMTP >>
```

```
C: quit
S: 221 Disconnection

The email has been received:
<RELAY>
Relay from: <master.com (193.1.124.54); 09 April 2003 09:02:13
o'clock IST; Version: 2.0> TO <master.org (193.1.124.51)>
</RELAY>
<HEAD>
From: bcrouzet@master.com
Send From: master.com (193.1.124.54); 09 April 2003 09:02:13 o'clock
IST; Version: 2.0
Date: 09 April 2003 09:02:13 o'clock IST
Message id: 1064375633560
To: 2@master.org
Subject: AMTP Procedure 3
</HEAD>
<BODY>
It is an AMTP procedure using RSET command.
</BODY>
```

6. Relay with Authenticated Mail Transfer Protocol

An open relay or relaying server is an email server that allows people to relay email. By processing email that is not for or from a local user, a relaying server makes it possible for an unscrupulous sender to route large volumes of spam email. The advantages of a relaying server are that a sender can send an email from anywhere outside his/her email network, a sender can use any application and the sender server does not have to know each recipient server to transfer an email. The drawback of a relaying relay is that spammers use it in order to hide their identity and the source of their personal computer. The relaying server only adds its information in the email header but it cannot verify the sender email address or insert any information concerning the sender identity.

With Authenticated Mail Transfer Protocol, the relaying server that allows people to relay email will do exactly the same transaction as a recipient server. It receives the notification for an email and retrieves the email. After this transaction, it will inform the recipient server that an email has to be retrieved on the relaying server. The email will arrive to the recipient even if the email passes by a relaying server. The transaction between the sender and the recipient within a relay will be longer than if there was a direct link between the two email servers. The advantages are that the spammer cannot use the relaying server to send anonymous and spoofed email and the sender server does not have to know each recipient server to transfer an email.

Sender <- AMTP -> Sender Server <- AMTP -> Relay Server <- AMTP ->
Recipient Server <- POP or IMAP -> Recipient

Figure 3: Presentation of transaction with a relay server

The sender server transfers an external email to the relaying server. When the route to send an email to the recipient is not known by the sender server, it goes through a relaying server to accomplish the transaction. The sender server enters into the Information state and informs the relaying server that an email is waiting to be retrieved. The relaying server accepts the email if it knows the recipient server or another relaying server that it can send the email to by checking its route table.

The relaying server enters into the Retrieved state and retrieves the email from the sender server. Instead of storing the email in the recipient mailbox, it stores the email as an external email and informs the recipient server or another relaying server about it. The email is stored into the relaying server and no longer in the sender server.

The relaying server continues with the Information state and waits for an answer from the recipient server. The recipient server checks and validates the recipient email address. If the recipient email address does not exist, it sends back an error message to the relaying server that sends an email to the sender to inform him/her about the fact that the recipient email address is incorrect. If the recipient email address is correct, the recipient server enters into the Retrieved state. It retrieves the email from the relaying server and stores the email in the recipient mailbox.

The transaction between a sender server and a relaying server is identical to a transaction between a sender server and a recipient server. The same transaction is also used between a relaying server and a recipient server. The difference with a relaying server is that the email has to be sent to another server. The relaying server will change the NUMBER parameter in the SELO command, by creating a new one to avoid a copy of the email.

The procedure of the AMTP relaying function is:

Step1:
Sender email Client --> Sender server
Using AMTP logout state, AMTP identified state and AMTP mail state

---

Step 2:
Sender server --> Relaying server

Sender server <-- Relaying server
Using AMTP Retrieved state

Step 3:
Relaying server --> Relaying server
Using AMTP information state

Relaying server <-- Relaying server
Using AMTP Retrieved state

Step 4:
Relaying server --> Recipient server
Using AMTP information state

Relaying server <-- Recipient server
Using AMTP Retrieved state

Step5:
Recipient server <-- Recipient email client
Using POP3 or IMAP transaction

Authenticated Mail Transfer Protocol is operational as a relaying server. The relaying server is used to transfer email to a recipient server. Authenticated Mail Transfer Protocol is therefore protected against anonymous and spoofed email. A user can send an email to his/her server and use the relaying server to transfer an email to other email servers.

7. Authenticated Mail Transfer Protocol Reply codes

7.1 New Reply Codes

Reply codes are important for a server and a user because it permits them to know if the transaction is correct or not. The reply code 555 informs the server for any errors that occur between two servers. The error permits the server to take action of it. There are four types of error: during the Identified state, during the transaction to send an email (Email State) and during the transaction between two servers for the commands SELO and SEMA.

For the Identified state, the reply codes are:
=> 503 Use the Command USER before other commands.
=> 401 User unknown û Enter the user information again - only 3 times.
=> 505 User does not exist û Connection close.
=> 250 User Accepted.

When the user sends an email, the reply codes are:

=> 501 The email is wrong.
=> 551 User not local.
=> 250 Server Ready.

When the server uses the command SELO, the reply codes are:
=> 555 Selo command error û Recipient Unknown, Argument missing, Command Unknown or Result Unknown.
=> 250 Selo Accepted.
=> 250 Mail accepted for delivery.

When the server uses the command SEMA, the reply codes are:
=> 555 Sema command error û Argument missing, Mail does not exist, Command Unknown or Result Unknown.
=> 555 Mail error.
=> 250 Sema Accepted.
=> 250 Mail delivered.

7.2 Reply Codes from Request For Comment 2821

Positive Completion replies are:
=> 211 System status or system help reply.
=> 214 Help message.
=> 220 Service ready.
=> 221 Service closing transmission channel.
=> 250 Requested mail action okay, completed.
=> 251 User not local.
=> 252 Cannot VRFY user, but will accept message and attempt delivery.

Positive Intermediate reply is:
=> 354 Start mail input; end with.

Transient Negative Completion replies are:
=> 421 Service not available, closing transmission channel.
=> 450 Requested mail action not taken: mailbox unavailable.
=> 451 Requested action aborted: local error in processing.
=> 452 Requested action not taken: insufficient system storage.

Permanent Negative Completion replies are:
=> 500 Syntax error, command unrecognized.
=> 501 Syntax error in parameters or arguments.
=> 502 Command not implemented.
=> 503 Bad sequence of commands.
=> 504 Command parameter not implemented.
=> 550 Requested action not taken: mailbox unavailable.
=> 551 User not local; please try.
=> 552 Requested mail action aborted: exceeded storage allocation.
=> 553 Requested action not taken: mailbox name not allowed.
=> 554 Transaction failed.

8. Test Carried Out Against Email Problems

8.1 Test Authenticated Mail Transfer Protocol Against Anonymous Email

The different tests on Authenticated Mail Transfer Protocol show that a user cannot send an anonymous email. The user was not able to change and to enter his/her email address during the process of sending an email. The old commands from SMTP (HELO, EHLO and MAIL FROM) do not work with AMTP. The user can only enter the recipient email address and his/her message. The user cannot modify the email in order to change or to enter a sender email address. The message is entered in the XML tag BODY and the sender server completes the email header using the XML tag HEAD. The sender server enters the sender email address into the email and sends the email at the end of the transaction to the correct recipient. Authenticated Mail Transfer Protocol stops anonymous email. This is the goal of the protocol.

The only possible way to send an email is to access the email server. The hacker has to create a file in the email server and a row in the email server database. Therefore, he/she can use the SELO command to inform the recipient server.

8.2 Test Simple Mail Transfer Protocol Against Anonymous Email

The test shows that it is possible to receive an email from an invalid email address from the same or a different domain name for the three external email servers and for the server prototype. When the user replies to the email, the server returns a delivery message from the sender server or an error message from the client interface. The email received contains some useful information that allows an expert to find an evidence of the sender. Examination of the line "Received From" in the email header makes the email traceable. Microsoft Exchange provides the IP address of the sender computer. Using a provider and a modem, the line "Received From" contains the IP address of the Internet provider used for the connection. Therefore, it is impossible to determine the address of the sender computer.

In addition to this, the email provider Free does not provide a complete email header: the line ôFromö is unspecified. Therefore, it is impossible to reply to the sender. It is possible to find a sender email address by looking at the email source: The field ôReturn-Pathö gives the user the sender email address. The web interface of the postfix server allows a user to modify his/her email address without finding any information about the true sender in the email header.

With Simple Mail Transfer Protocol, it is possible to send an anonymous email with an external or internal sender email address. Different countermeasures exist such as time out, different email

servers or a web server. These tests are simple and stay in the

protocol level. A hacker will study the email server to see if he/she can use it. These tests still prove that Simple Mail Transfer Protocol is insecure. In addition to this, email servers do not respect the email format. Therefore, different applications such as Outlook Express or Microsoft Outlook cannot use the email header.

8.3 Test Authenticated Mail Transfer Protocol Against Spoofed Email

The different tests on Authenticated Mail Transfer Protocol show that a user cannot send a spoofed email. The user was not able to change or to enter his/her email address during the process of sending an email using the old commands of SMTP: EHLO, HELO or MAIL FROM. The user has to enter the recipient email address and his/her message. He/she cannot enter a sender email address in the email header or use another email to change the sender email address. Authenticated Mail Transfer Protocol stops spoofed email and is the goal of this protocol.

There are two possible ways to send a spoofed email. The first possibility is to find a username and a password; therefore, a user can send a spoofed email. The second way is to access the email server in order to create a file in it and a row in its database. Therefore, a user can use the SELO command to inform the recipient server about an email created by him/her waiting to be retrieved.

8.4 Test Simple Mail Transfer Protocol Against Spoofed Email

The test shows that it is possible to receive an email from someone elseÆs email address from the same or a different domain name for the three external email servers. The sender did not send this email but someone else did. All sender email addresses used for this test are valid. Therefore, there is no need to send a reply to the sender. For all email servers, the server accepts any sender email address without verifying the domain name of the sender. The web interface of the Postfix server shows that it is possible to send an external email using someone elseÆs email address without any information in the email header about the true sender. The only information is the header field "X-Originating-IP". These tests prove that Simple Mail Transfer Protocol is insecure.

8.5 Test Authenticated Mail Transfer Protocol Against Relaying Email

The test shows that it is possible to use a relaying server in order to send an external email with Authenticated Mail Transfer Protocol. The sender sends his/her email to the sender server. The sender server transmits the email to the relaying server using the Information and Retrieved states. The relaying server relays the email to the recipient server using the Information and Retrieved states. The recipient server retrieves and copies the email into the recipient mailbox. The recipient is able to read his/her email using

the protocol POP3 implemented on the server prototype. In the email,

the XML tag RELAY contains all the relay information in order to be able to trace the route of the email and to find all email server addresses.

8.6 Test Simple Mail Transfer Protocol Against Relaying Email

The test shows that it is possible to send an email to an external recipient using a relaying server on condition that the instructions of the RFC 821 (Simple Mail Transfer Protocol) are followed. The transaction between a client and a server is identical to the transaction between a server and a server. Simple Mail Transfer Protocol allows a sender to relay an email from any email server.

This test demonstrates that a real email server is not able to relay an email and to send an external email. The user has to use a Graphical User Interface provided by the email server to send an external email. Administrators have turned off the relaying function in order to protect the user against spam email. Microsoft Exchange, the Mail Transfer Agent Exim and Postfix do not authorise any user to send external email. This test has been conducted with a command prompt using a telnet connection and the protocol SMTP.

A simple study of the server information provided by any of these email servers does not allow a user to find an access to the email server. Therefore, it is difficult for an external user to send an email outside his/her network. Current email servers provide a Web interface in order to solve this problem but a programmer cannot use his/her client interface in order to send an email.

9. Test Carried Out in a Network

9.1 Presentation

In order to protect a network, it is possible to use a router, a firewall, a proxy server or a firewall associated with a proxy server. It is important to accept or refuse requests coming from outside the network or leaving the network. A network has to be protected in order to increase the security of the user data. The following figure represents one possibility for protection of a network. Network 1 is outside Network 2. The router, the firewall or the proxy server is used as a gateway that allows Network 2 to establish a route to Network 1. When an administrator combines these protections, the diagram for the network is different. In any case, an administrator needs a gateway to connect Network 1 with Network 2. It is possible to have a firewall, a proxy server and a router in Network 2 in order to increase the security of the network.

Figure 3: Presentation of protections for the network

The first protection is a router. A router is a physical device that joins multiple networks together in order to form the World Wide Web. Routers have the ability to filter traffic, either incoming or outgoing based on the IP addresses of senders and receivers. The Access List in a router is used to ban or to authorise some packets to enter the network. The Access list has to be configured in the router configuration.

The second protection is a firewall. A firewall is used to filter IP packets going into, or coming out of the network. A firewall can block, forward or pass the packet to the final recipient. The firewall can be set up to filter a protocol (i.e. TCP, UDP or ICMP), a port, an IP address or a range of IP addresses. A firewall is the most powerful tool to filter the packets from the network but not to protect the IP address of the network.

A firewall is a system or combination of systems that enforces a boundary between two or more networks and keeps intruders out of internal networks. Firewalls serve as barriers for packets passing from one network to another. The command IPCHAINS [5] creates a firewall under Linux. The software ôSolidShare 2.0ö [6] is used as a firewall under Windows. The configuration of these two firewalls is to accept TCP connections and refuse UDP and ICMP connections.

The last protection is a proxy server. A proxy server is used to filter the packet going into, or coming out of the network. It is similar to a firewall but the proxy server will keep the network completely inaccessible from outside the network. The proxy server redirects any queries (HTTP, SMTP or FTP) to the server in charge of the protocol whether it is inside or outside the network. From an outside point of view, the user believes the proxy server is the server in charge of the network. He/she cannot establish a connection to any server inside the network except the proxy server. In order to establish a transaction going out of the network, the user establishes a connection with the proxy server and then the proxy server requests the userÆs queries. The proxy server changes the user IP address in the packet and replaces it by its IP address.

A proxy is a system that allows a network to share a single Internet connection. The shared Internet connection can be anything from a dialup modem to a dedicated T1 circuit. Under Linux, the proxy server is ôTCPProxy 1.1.6ö [7] and under Windows, the proxy server is ôGateKeeper Pro 4.5ö [8].

An administrator can combine these protections and obtain a well-protected and secured network. The challenge for him/her is to find

the right configuration that protects every server and computer, and

allows the user to have access to all data authorised outside and inside the network.

9.2 Test the Transfer Protocol with Routers as a gateway

The test consists in sending an email with SMTP or AMTP between two email servers through routers (CISCO 2600 and CISCO 2500). The test is conducted with the prototype on an internal network. Four tests were performed with one, two, three and four routers. In each test, the transaction with the two protocols was carried out. The IP address of the sender address (192.5.5.10) is identical for the series of tests. The IP address of the recipient server changes according to of the test. The four routers have been configured to accept the transfer of all packets in any direction.

The result of the test shows that Authenticated Mail Transfer Protocol and Simple Mail Transfer Protocol are operational behind routers. For the series of tests, these two protocols are able to transmit and receive information through routers. The sender server communicates through different routers and finds the route of the recipient server. Therefore, AMTP and SMTP can access the World Wide Web with no difficulty.

9.3 Test the Transfer Protocols with a Firewall as a gateway

The result of the test shows that Authenticated Mail Transfer Protocol and Simple Mail Transfer Protocol are operational behind a firewall. For the series of tests, these two protocols were able to transmit and receive information through a firewall.

9.4 Test the Transfer Protocols with a Proxy Server as a gateway

A proxy server replaces the IP address of the packet before it is forwarded to the email server. The goal of the proxy server is to avoid people learning the address of any computer inside the local network. SMTP is operational behind a proxy server because the protocol does not verify the IP address of the packet and does not use the IP address of the sender server.

An AMTP server does not work behind a proxy server because the proxy server changes the IP address in the packet but not the IP address of the SELO command. Therefore, the recipient server is unable to establish connection with the sender server. Two solutions can be implemented: The AMTP server runs on the same computer as the proxy server OR the programme does not check the IP address of the packet behind a proxy server. The first solution for AMTP is to have the email server on the same computer as the proxy server. Therefore, it is the same test as to send an external email on the same network. The test was carried out and showed that it works. The second solution for AMTP is to modify the protocol in order not to compare the IP address of the packet with the IP address of the SELO

command. The programme enters the IP address of the proxy, and it

can execute the Retrieved state but the level of security of the protocol is decreased. These two solutions allow an AMTP server to work behind a proxy server.

9.5 Test the Transfer Protocols with a Firewall associated with a Proxy Server as a gateway

The result of the test shows Simple Mail Transfer Protocol and Authenticated Mail Transfer Protocol are operational behind a firewall associated with a proxy server. The firewall lets the packet go to the email server without changing the IP address of the packet. The proxy server does not touch the packet. For the series of tests, these two protocols were able to transmit and receive information through a firewall.

9.6 Test the Transfer Protocol in a World Wide Web Simulation

The result of the test shows that Authenticated Mail Transfer Protocol and Simple Mail Transfer Protocol are operational on the World Wide Web. For this series of tests, these two protocols were able to transmit and receive information through routers, a firewall and a relaying server.

10. Comparison of the Speed of the Two Transfer Protocols

The test consists in comparing the speed of the two transfer protocols: Authenticated Mail Transfer Protocol and Simple Mail Transfer Protocol. The server prototype measures the time that the server or the client interface needs to perform a transaction. The user employs the client prototype to send one, five, ten and twenty emails with two different sizes of data through ten different configurations. The small data size contains 472 bytes and the large data size contains 6085 bytes. The average total of email represents the average to send 35 emails (5 + 10 + 20). The first email sent initiates the route in order to establish a connection to the recipient server. Therefore, the result is not taken into consideration in the average total.

The ten different configurations are:
C1: The user sends an internal email.
C2: The user sends an external email on the same network.
C3: The user sends an external email using an email server as a relay between two email servers.
C4: The user sends an external email using a firewall under Linux between two email servers.
C5: The user sends an external email using a firewall under Windows between two email servers.
C6: The user sends an external email using one router between two email servers.
C7: The user sends an external email using two routers between two email servers.

C8: The user sends an external email using three routers between two email servers.
C9: The user sends an external email using four routers between two email servers.
C10: The user sends an external email using four routers and a relaying server: Simulation of a World Wide Web

The following table displays the time in milliseconds and represents the average for sending 35 emails sent using a small or large data size for Authenticated Mail Transfer Protocol (AMTP) and Simple Mail Transfer Protocol (SMTP). The time takes into consideration the Client-to-Server and the Server-to-Server transaction times. The field Difference represents the difference between AMTP and SMTP (time with AMTP û time with SMTP). It compares the times in order to decide which one is the quickest to transfer an email. In brackets, we have the quickest protocol.

Name	Small Size of Data			Large Size of data		
	AMTP	SMTP	Difference	AMTP	SMTP	Difference
C1	225	172	53 (SMTP)	221	197	25 (SMTP)
C2	1455	894	561 (SMTP)	1345	905	440 (SMTP)
C3	3442	2912	530 (SMTP)	3193	2172	1022 (SMTP)
C4	9762	5459	4303 (SMTP)	9834	6167	3667 (SMTP)
C5	9925	5299	4626 (SMTP)	10019	5159	4860 (SMTP)
C6	1287	838	449 (SMTP)	1305	1071	234 (SMTP)
C7	1510	1721	-211 (AMTP)	4255	7029	-2774(AMTP)
C8	2090	2446	-356 (AMTP)	4769	7847	-3077(AMTP)
C9	2087	3447	-1360(AMTP)	2063	2753	-690 (AMTP)
C10	3446	5698	-2252(AMTP)	9848	8334	1513 (SMTP)
Total	35004	28713	6290 (SMTP)	46632	41437	5195 (SMTP)

All tests have been realised with the same prototype using the same algorithm to send an email. As expected, Simple Mail Transfer Protocol is quicker than Authenticated Mail Transfer Protocol. Overall, the results show only a small difference between these two transfer protocols. Authenticated Mail Transfer Protocol takes 5.2 and 6.3 seconds more to send an email to a recipient than Simple Mail Transfer Protocol. AMTP is better in three cases: Configuration C7, C8 and C9 (routers tests). For a small data size, AMTP is better than SMTP for the configuration C10 (WWW). For other configurations, SMTP is better.

During the test, a problem occurred for configuration C3 with AMTP: the server prototype produces some database errors; it tries to

insert a row with the same primary key. Because of this, an error

occurs and the processing time increases. This problem was solved with a random number for the primary key but some errors still happen. The server prototype looses time finding a primary key for the EXTQUEUE table needed to verify the parameters of the SEMA command.

It is possible to reduce the time for a Client-to-Server transaction for AMTP. In the server prototype, AMTP sends three times more information than SMTP, i.e. SMTP sends only one line while AMTP sends three lines for each reply. Therefore, the time taken to send these two lines increases the average time. Two tests were conducted with a reduction of the Client-to-Server transaction using the configuration C1 (internal mail) and C2 (external mail with a direct link to the recipient server). The server prototype sends only one line between each command.

Name	Small Size of Data			Large Size of data		
	AMTP	SMTP	Difference	AMTP	SMTP	Difference
C1	164	202	-38 (AMTP)	171	198	-27 (AMTP)
C2	610	807	-198(AMTP)	634	739	-105 (AMTP)
Total	774	1009	-235(AMTP)	806	938	-132 (AMTP)

The result of this test shows that AMTP is quicker for each configuration. This simple test demonstrates that AMTP can be quicker than SMTP for all configurations if the Client-to-Server transaction is shorter. This transaction is shorter on protocol but longer on server prototype in order to explain the new transfer protocol to the user.

In the future, the series of tests should send more emails and should be conducted on a real network with two email servers located on different places. The Client-to-Server procedure should be reduced on the server prototype. This series of tests should give a better demonstration of the efficiency of AMTP.

11. Attack against an Authenticated Mail Transfer Protocol server

The two server commands, SELO and SEMA, can be used as an attack against the email server and allow a hacker to obtain a user email address. A hacked needs more resources in order to do so. He/she has to run an Authenticated Mail Transfer Protocol server on port 26. It is more difficult for him/her because only one programme per server can listen to port 26. A hacker cannot implement a programme on an existent Authenticated Mail Transfer Protocol server. Moreover, he/she cannot use a telnet connection to send an anonymous email or

create a fake Authenticated Mail Transfer Protocol server on a

computer without port 26. If a hacker has his/her own server, his/her IP address is in the packets and identifiable by the administrator.

A hacker can use the server command in the protocol in order to attack the server. If he/she wants to succeed, he/she has to know the message number and the recipient email address, which he/she cannot have both at the same time. If a hacker tries too many times, the server discovers the attack and closes the connection. A hacker does not affect a user, only the performance of the server. Two commands are available for him/her: SELO and SEMA. The SELO command is used to inform a recipient server that an email has to be retrieved. The SEMA command is used to retrieve an email.

11.1 Using the SELO command

The hacker has to create an email inside the email server and a row in the email database. He/she informs the recipient server which server tries to retrieve the email and it will succeed only if the hacker has created the two conditions: email inside the server and a row in the database. Using the SELO command, the hacker can obtain a valid email address from the server. The SELO parameters for this test are:
The name or IP address of the sender server is 193.1.124.54
The email number is 123456789
The recipient email address is:
Valid Internal email address: bcrouzet@master.com
Invalid Internal email address: anonymous@master.com
External email address: anonymous@master.org

The server returns an error because the email does not exist. The error SAAN3 specifies ôThe function semaAction cannot find the filename and the sender in the database: table extqueueö. Therefore, the sender server cannot deliver the email to the recipient server.

On the server prototype, the email server displays the error SA10 that specifies that the reply code is different from 250. The server is informed that an email with the number 123456789 for anonymous@master.org is waiting on the server 193.1.124.54 to be retrieved. The recipient server will try to retrieve this email using the SEMA command.

The server returns an error to specify that the user does not exist locally. The server will not try to use the Retrieved state. The problem remains that a hacker is able to find a valid recipient email address from the server and to trigger the Retrieved state. Whatever the situation, the hacker was not able to send an email. The only protection against his/her attack is to have a list of IP addresses that uses the SELO command. If the IP address is repeated

many times and the server does not use the Retrieved state, then the

administrator has to ban the IP address or check the problem. Any email server does not allow people to establish a connection on port 23 (telnet 23). Therefore, it is difficult to insert data into an email server.

11.2 Using the SEMA command

The hacker can use the SEMA command to retrieve an email that it is not for him/her. In order to do this, he/she needs to find the two parameters of the SEMA command that correspond to an email on the sender server. The SEMA command needs 2 parameters:
The recipient email address is: bcrouzet@master.com
Number: 123456789

The result of the SEMA command is identical for all email addresses. The server does not have the valid information in its database. The server returns an error because the email does not exist. The error SAAN3 specifies ôThe function semaAction cannot find the filename and the sender in the database: table extqueueö. Therefore, the sender server cannot deliver the email to the recipient server. The user bcrouzet@master.com did not receive an email and the hacker did not retrieve an email from a user mailbox.
The test shows that the hacker was able to do nothing. If the number and the recipient email address are incorrect, the server returns an error. Again, a list of IP addresses will be helpful to protect the server against this type of attack. Any failed SEMA command has to be studied in order to understand the error. Whatever the situation, the hacker was not able to retrieve an email.

11.3 Overview of attacks against an Authenticated Mail Transfer Protocol server

Authenticated Mail Transfer Protocol provides two server commands that are used during the Server-to-Server transaction. Hackers can used these two commands in order to hack the system. The different tests on Authenticated Mail Transfer Protocol show that a hacker can find a local email address on the email server but he/she cannot access the email server or retrieve an email. To find the number and the recipient email address is a very difficult task. These two parameters depend on the sender server and the user. It is possible to find the algorithm that produced the number but it will be difficult to find the recipient email address and the number, both at the same time. The recipient email address depends on the sender and the number depends on the number of messages sent. These numbers are stored in the server, where it is difficult to crack the database. In addition to this, it is important to maintain a list of IP addresses, which contains failed SELO or SEMA commands in order to determine the address of the attacker.

12. Conclusion

In conclusion, this is one solution investigated to recognise a user by a server. The Authenticated Mail Transfer Protocol server identifies the user before he/she can use the email server. The drawback is that the transaction between two email servers takes more time than Simple Mail Transfer Protocol. This solution offers the guarantee that the sender exists and avoids spoofed and anonymous email, which is the aim of the exercise. It is a major step forward in the success of the thesis.

The sender server identifies any sender and the recipient server is able to trust the information provided by the sender server. The recipient trusts the sender email address and he/she can find the sender without any difficulty. This solution does not stop spam email but it reduces the negative effect of this email problem. A user has the possibility of locating the sender and informing the sender server.

Authenticated Mail Transfer Protocol modifies the structure of an email in order to find essential information in the email faster. It also improves the relaying function in order to use it. It adds new commands and improves the service offered during a transaction.

The advantage of this solution is the difficulty for a hacker to find the number and the recipient email address from the SELO and SEMA commands. These two parameters depend on the sender server and the sender. The recipient email address depends on the sender and the number will depend on the number of email sent. These numbers are stored in the sender server, where it is difficult to crack the database. It is possible to find the algorithm that produced the number but it will be difficult to find the recipient email address and the number together.

This document has demonstrated shown that Authenticated Mail Transfer Protocol is more secure and advanced than Simple Mail Transfer Protocol. The series of tests demonstrates that Authenticated Mail Transfer Protocol solved two email problems: anonymous and spoofed email and improved the relay between email servers.

In the local network, Authenticated Mail Transfer Protocol works behind routers and gateways except proxy servers. Authenticated Mail Transfer Protocol is operational to work on a real World Wide Web. Authenticated Mail Transfer Protocol is slower than Simple Mail Transfer Protocol but it only takes 5-6 seconds more than Simple Mail Transfer Protocol in the overall result. A reduction of the Client-to-Server transaction on the prototype shows that Authenticated Mail Transfer Protocol outperforms Simple Mail

Transfer Protocol. Different configurations help to show that

Authenticated Mail Transfer Protocol works as fast as Simple Mail Transfer Protocol.

Hackers can attack Authenticated Mail Transfer Protocol with two server commands, SELO and SEMA, without damaging the email server. Authenticated Mail Transfer Protocol offers different advantages with three XML tags (RELAY, HEAD and BODY) in the structure of the email and two new commands (HEAD and MORE TO). In conclusion, Authenticated Mail Transfer Protocol outperforms Simple Mail Transfer Protocol and offers to users different advantages that will improve daily use.

Security Considerations

Security Considerations has been described during this document.

References

[1] Postel, J. (1982) Simple Mail Transfer Protocol. RFC 0281. ISI

[2] Klensin, J. (2001) Simple Mail Transfer Protocol. RFC 2821. AT&T Laboratories.

[3] Crocker, D. (1982) Standard for the format of ARPA Internet text messages. RFC 0822. University of Delaware.

[4] Resnick, P. (2001) Internet Message Format. RFC 2822. QUALCOMM Incorporated.

[5] Russell, R. (2000) Linux IPCHAINS-HOWTO [Online]. Available from: http://www.tldp.org/HOWTO/IPCHAINS-HOWTO.html [Accessed 04 September 2002].

[6] Lijun, Q. (2002) Internet sharing and firewall - SolidShare [Online]. Available from: http://www.solidshare.com [Accessed 04 September 2002].

[7] Zekoll, W. (2002) tcpproxy - Generic TCP/IP Proxy [Online]. Available from: http://www.quietsche-entchen.de/software/tcpproxy.html [Accessed 04 September 2002].

[8] Infopulse (2002) Proxy - Pro GateKeeper - Your Internet Sharing Solution - Proxy Server [online]. Available from: http://www.proxy-pro.com/index.html [Accessed 04 September 2002].

[9] Postel, J. (1981) Internet Protocol - DARPA Internet Program Protocol Specification. RFC 791. USC/Information Sciences Institute.

[10] Postel, J. (1981) Transmission Control Protocol - DARPA Internet Program Protocol Specification. RFC 793. USC/Information Sciences Institute.

[11] RFC Editor. (2001) Definition of RFC [Online]. Available from: http://www.rfc-editor.org/overview.html [Accessed 10 December 2001].

[12] Freed, N. Borenstein, N. (1996) Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. RFC 2045. Innosoft. First Virtual.

[13] Crocker, D., Overell, P. (1997), Augmented BNF for Syntax Specifications: ABNF. RFC 2234. Internet Mail Consortium. Demon Internet Ltd.

Appendix

Appendix A: Acronyms

ASCII=> American Standard Code for Information Interchange (ASCII) is the most common format for text files in computers and on the Internet. In an ASCII file, each alphabetic, numeric, or special character is represented with a 7-bit binary number (a string of seven 0s or 1s). 128 possible characters are defined [13].

IP => Internet Protocol (IP) is designed for use in interconnected systems of packet-switched computer communication networks. The IP provides for transmitting blocks of data called datagramÆs from sources to destinations, where sources and destinations are hosts identified by fixed length addresses. The IP also provides for fragmentation and reassemble of long datagramÆs, if necessary, for transmission through "small packet" networks [9].

MIME => Multipurpose Internet Mail Extensions (MIME) is an extension of the original Internet email protocol that lets people use the protocol to exchange different kinds of data files on the Internet. The type of data can be audio, video, images, application programs, and other kinds, as well as the ASCII handled in the original protocol (SMTP) [12].

RFC => Request For Comment (RFC) forms a series of notes, started in 1969, about the Internet. The notes discuss many aspects of computer communication, focusing on networking protocols, procedures, programmes, and concepts but also including meeting notes, opinion, and sometimes humour [11].

SMTP => Simple Mail Transfer Protocol (SMTP) is to transfer any mail from a client to a server and is defining in RFC 0821 [1] and RFC 2821 [2]. The protocol used the port 25 to receive the data and the TCP/IP protocol to transport the data in the network.

TCP => Transmission Control Protocol (TCP) is intended for use as a highly reliable host-to-host protocol between hosts in packet-switched computer communication networks, and in interconnected systems of such networks [10].

Appendix B: Terminology

=> A mail, email, message or electronic mail represents a message sent across the network from one person to another.
=> Anonymous email is email that has been directed to a recipient through a third-party server that does not identify the originator of the message.
=> Client refers to the user software.
=> Command represents a specific order from a user to an application to perform a service.
=> Hacker is a person who tries to break into the computer system.
=> Mail Agent System represents a system to manage the mail (write, read, delete and send).
=> Authenticated Mail Transfer Protocol characterises the Simple Mail Transfer Protocol version 2.
=> Protocol or standard represents a set of rules for a subject.
=> Recipient represents the user who receives a mail and is in the server side.
=> SA represents a SMTP server where the sender is known.
=> SB represents a SMTP server where the recipient is located.
=> Sender represents the user who sends a mail and is in the client side.
=> Server represents the application running on the server side.
=> Spam is unsolicited email on the Internet.
=> Transaction is an exchange of information between 2 servers or a server and a user.
=> User is used to refer to a human user.

Author's Addresses

Brice Crouzet (PK4)
Institute of Technology Tallaght
Tallaght
Dublin 24
Ireland

Phone: + 353 (0) 14 04 23 45
Fax: + 353 (0) 14 04 20 00
E-mail: brice.crouzet@it-tallaght.ie

Copyright Notice

others, and derivative works that comment on or otherwise explain it