Independent Submission X.Chen
Request for Comments: xxxx
Category: Informational December 2017

Mail Agent Protocol (MAP)

Status of This Memo

This document specifies the detail of a mail agent protocol, which can be used to send e-mail to recipient via e-mail address. This is a term project for CSCI 6431 computer network.

Abstract

This document is a specification of design and implementation of the mail agent protocol which can send e-mail, including subject and character content, to recipient through the e-mail address. The e-mail transmission mechanism of this agent is based on the RFC 821-Simple Mail Transfer Protocol.

Copyright Notice

Table of Contents

1.Introduction

One of the most prevalent modern web service is undoubtedly the e-mail service, millions of Internet users are using e-mail every day for work, study or chat. What's more, the e-mail greatly facilitates our way of communication. In the past, people must wait months or even years to receive their mail, however, with the help of the e-mail, Alice and Bob, who live in different corners on the earth can transfer their mail to each other within a second.

Therefore, it is important and interesting to get familiar with this useful service, e-mail, which is basically built on the SMTP protocol. As an old saying goes, practice makes perfect. Thus, the best way to understand the SMTP protocol is building a mail agent that can send mails. Its implemented in the protocol named mail agent protocol(MAP).

2. The MAP Model

2.1. Architecture

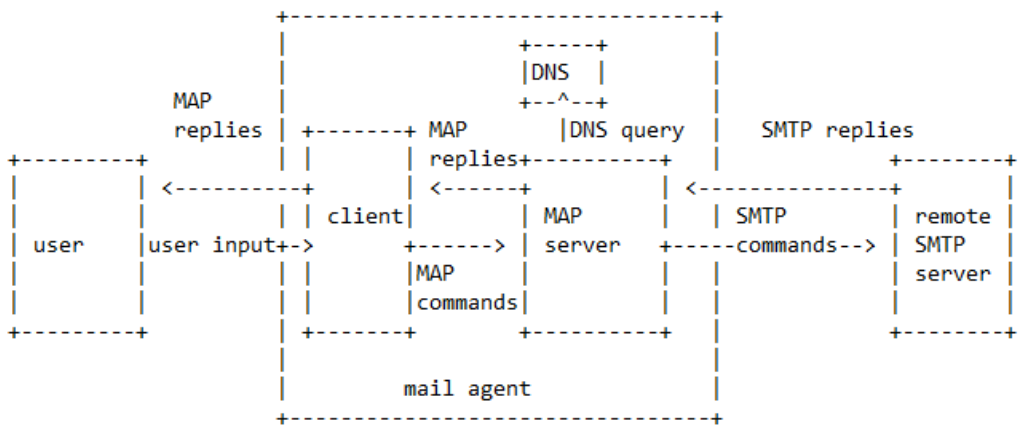
The architecture of the mail agent protocol is client-server(CS) architecture. The client is used to collect user input and transfer it to the MAP server within the agent via MAP commands. The MAP server serves as an intermediate between the client and the remote SMTP server, it starts up when the client is open and shuts down when the client is terminated. The responsibility of the MAP server is to establish a connection with the remote SMTP server, convert received MAP commands to corresponding SMTP commands and communicate with the remote SMTP server.

2.2. Socket

The mail agent needs an interface to send data into the Internet. The Socket serves as an endpoint of a 2-way communication link between the agent and the remote SMTP server, it helps the agent to send and receive data. The port that the mail agent uses to communicate with the remote SMTP server is port 25 and the hostname depends on the recipient's e-mail address (e.g. SMTP server hostname for 407184033@qq.com is mx1.qq.com)

2.3. Basic Structure

The MAP design is pictured as follow:



When user wants to send an e-mail via the mail agent, the first step is to input the information into the client through the graphic user interface (GUI) provided by the client. If the format of information (e.g. mistaken format of email address) is incorrect, the client will generate corresponding MAP replies, otherwise, the client gathers the information and send it to the MAP server via MAP commands.

Once the MAP server receives MAP commands, it performs DNS query based on the mail address user provided, and respond with corresponding MAP reply. During the communication to the remote SMTP server, the MAP server will respond with MAP replies to client depend on the corresponding SMTP replies it received from the remote SMTP server. Additionally, client will show these replies to the user.

2.4. Client GUI Design

There will be three rows in the GUI for the input of sender's e-mail address, the recipient's e-mail address and subject respectively. A large plain text area is provided for the input of the mail content. The input is restricted to the ASCII character set (the standard of the SMTP protocol). At the bottom of the GUI, there will be a button named "Send", by clicking it, the client will proceed to send the information to the MAP server.

Furthermore, there will be pop-ups to show the MAP replies. These replies may be generated by client itself or received from the MAP server.

3.The MAP Specifications

3.1. MAP Commands

The map commands are used to transfer the segmented e-mail information from the client to the MAP server, which will eventually convert the MAP commands to corresponding SMTP commands. The commands themselves are alphabetic characters. They start with "CMD_" and end with a period.

3.2. Command Syntax and Semantics

1."CMD_FROM". This command is used to send the e-mail address of the sender to the MAP server.

Syntax: "CMD_FROM: <sender's e-mail address>."

2."CMD_TO". This command is used to send the recipient's e-mail address to the MAP server.

Syntax: "CMD_TO: <recipient's e-mail address>."

3."CMD_SUBJECT". This command is used to send the subject of the e-mail to the MAP server.

Syntax: "CMD_SUBJECT: <subject string>."

4."CMD_CONTENT". This command is used to send the content of the e-mail to the MAP server.

Syntax: "CMD_CONTENT: <content string>."

5.“CMD_END”. This command is used to inform the MAP server the end of the e-mail.

Syntax: “CMD_END.”

6. “CMD_SMTP_CONN”. This command is used to establish connection with the remote SMTP server.
Syntax: “CMD_SMTP_CONN.”

6. “CMD_SMTP_HELO”. This command aids the MAP server in the connection to the remote SMTP server.

Syntax: “CMD_SMTP_HELO.”

3.3. Commands for MAP Command Conversion

Following commands are used to convert the MAP commands into corresponding SMTP commands to facilitate the communication between the MAP server and the remote SMTP server.

1. “CMD_SMTP_FROM”. This command is used to transfer message in CMD_FROM into the field of SMTP FROM command.

Syntax: “CMD_SMTP_FROM.”

2. “CMD_SMTP_TO”. This command is used to transfer message in CMD_TO into the field of SMTP TO command.

Syntax: “CMD_SMTP_TO.”

3. “CMD_SMTP_DATA”. This command is used to transfer message in CMD_SUBJECT an CMD_CONTENT into the field of SMTP DATA command.

Syntax: “CMD_SMTP_DATA.”

3.4. MAP Replies

The MAP replies are used to indicate the status of the e-mail transmission that whether the transmission is successful or not, exceptions like invalid e-mail address format and translation of reply codes from the remote SMTP server. The MAP server will not respond unless there is an

exception, or the transmission success. These alphabetic characters start with “RE_” and end with a period.

3.5. Reply Syntax and Semantics

1.“RE_NO_EMPTY”. This reply is used to indicate that section of e-mail (e.g. e-mail content) is empty, which is not allowed.

Syntax: “RE_NO_EMPTY: <location of the empty occurrence>.”

2.“RE_INVALID_FORMAT”. This reply is used to indicate the format of the e-mail address is not correct.

Syntax: “RE_INVALID_FORMAT: <invalid e-mail address>.”

3. “RE_DNS_FAIL”. This reply indicates that the domain name provided in the recipient’s e-mail is not valid, that is, the type MX DNS query returns no record.

Syntax: “RE_DNS_FAIL: <invalid domain>.”

4.“RE_SERVER_UNREACH”. This reply will be generated when or 220 SMTP reply code from the remote SMTP server is not received. Additionally, it can be caused by network breakdown

Syntax: “RE_SERVER_UNREACH.”

5.“RE_PORT_BLOCKED”. This reply is used to indicate that the port 25 on the localhost is blocked, either by operating system or ISP.

Syntax: “RE_PORT_BLOCKED.”

6.“RE_BADREPLY”. This reply is used to indicate that the MAP server does not receive expected SMTP reply code from the remote SMTP server.

Syntax: “RE_BADREPLY: <bad SMTP reply code>.”

7.“RE_SEVICE_DENIED”. This reply is used to indicate the failure of the e-mail transmission. The reply can be caused by a requirement of static IP address to pass the spam filter of the remote SMTP server or receiving 550 reply code.

Syntax: “RE_SERVICE_DENIED.”

8.“RE_SUCCESS”. This reply is used to indicate the success of the e-mail transmission. It will be generated when MAP server receives 221 SMTP reply code from the remote SMTP server.

Syntax: “RE_SUCCESS.”

4. The MAP Procedure

4.1. Client Session

When user starts the MAP agent, a new MAP session starts. A GUI will show up, waiting for user inputs. User fills the text fields with corresponding message and clicks the Submit button when it is done.

Once the client captures the click action, it starts convert message in text field into MAP commands and begins to check the empty or incorrect field. For instance, following field of CMD_FROM is of wrong format, the client will generate MAP reply.

GUI: CMD_FROM: <chenxuanweigmail.com>.

Client: RE_INVALID_FORMAT: <chenxuanweigmail.com>.

After verification of all of the MAP command, the commands will be delivered to the MAP server.

4.2. Server Session

When MAP commands are delivered, the MAP server first begin to check availability of the port 25. Then it performs dns query for the hostname in the field of CMD_TO. If the dns query for the specific hostname is failed, the server will generate a MAP reply.

Client: CMD_TO: <chenxuanweigmail.com>.

Server: RE_DNS_FAIL: <invalid domain>.

If everything proceeds without exception, the map server will establish a Socket connection with remote SMTP server over port 25. Then, the server converts the MAP command into corresponding SMTP command and send it to the remote

SMTP server. During the transmission, if the reply code is not what MAP server expected, it will generate corresponding MAP reply. For instance, a 550 SMTP code is received from the remote SMTP server.

Remote SMTP server: 550.

Server: RE_SERVICE_DENIED.

When all e-mail content is delivered to the remote SMTP server, the MAP server will automatically generate a SMTP quit command to the remote SMTP server and once it receives 221 SMTP code, it will yield a MAP reply to indicate the success of the e-mail transmission.

Remote SMTP server: 221.

Server: RE_SUCCESS.

5. Potential Problems

There are some limitations in this mail agent protocol. First, as an application layer protocol that interact with the SMTP protocol, it needs to establish a connection with remote SMTP server over port 25. However, some ISP such as Verizon has blocked the port 25, making this protocol unable to work normally. An alternative approach is to utilize AT&T mobile network which has port 25 opened.

Additionally, most mail service providers such as Hotmail or Gmail have an advanced spam filter which will reject e-mail that sent from a dynamic IP, that is, they will return a 550 SMTP reply code to the mail agent. A possible solution is to send the e-mail via a static IP, but the price is quite expensive. For instance, it would cost $500 from Verizon.

6. Security Considerations

This document tries to clarify the descriptions of the status of an application layer protocol, mail agent protocol(MAP). Misunderstanding the status of a memo could cause interoperability problems, hence security and stability problems.

7. References

[1] Postel, J., "Simple Mail Transfer Protocol", STD 10, RFC 821,August 1982.

[2] Klensin, J., Ed., "Simple Mail Transfer Protocol", RFC 2821, DOI 10.17487/RFC2821, April 2001.

[3] Daigle, L., Ed., Kolkman, O., Ed., and IAB, "RFC Streams, Headers, and Boilerplates", RFC 5741, DOI 10.17487/RFC5741, December 2009

[4] Kurose, J. F., & Ross, K. W. (2017). Computer networking: a top-down approach. Boston: Pearson.

Appendix A. Scenarios

A.1. A Typical MAP Transaction

This MAP example shows how the MAP protocol works when a user from cxw@apple.com wants to send an e-mail to 407184933@qq.com with subject “test” and content “test content.”. C, S and SS represent client, MAP server and remote SMTP server respectively.

C: CMD_FROM: <cxw@apple.com>.

C: CMD_TO: <407184933@qq.com>.

C: CMD_SUBJECT: <test>.

C: CMD_CONTENT: <test content.>.

S: CMD_SMTP_CONN.

SS: 220 newusamx48.qq.com MX QQ Mail Server

S: CMD_SMTP_HELO.

SS: 250 newusamx48.qq.com

S: CMD_SMTP_FROM.

SS: 250 Ok

S: CMD_SMTP_TO.

SS: 250 Ok

S: CMD_SMTP_DATA.

SS: 354 End data with <CR><LF>.<CR><LF>

SS: 250 Ok: queued as

S: CMD_SMTP_QUIT.

SS: 221 Bye.

S: RE_SUCCESS.

A.2. Empty Input

This example shows how MAP respond to the empty input.

C: CMD_FROM: <cxw@apple.com>.

C: CMD_TO: < >.

C: CMD_SUBJECT: <test>.

C: CMD_CONTENT: < >.

C: RE_NO_EMPTY: <To, Content,>.

A.3. Invalid Email Address

This example shows how MAP respond to an invalid email address.

C: CMD_FROM: <cxw@apple.com>.

C: CMD_TO: <407184933qq.com>.

C: CMD_SUBJECT: <test>.

C: CMD_CONTENT: <test content.>.

C: RE_INVALID_FORMAT: <407184933qq.com>.

A.4. Invalid Domain Name

This example shows how MAP respond to an invalid email address.

C: CMD_FROM: <cxw@apple.com>.

C: CMD_TO: <407184933@qqqqq12372.com>.

C: CMD_SUBJECT: <test>.

C: CMD_CONTENT: <test content.>.

S: RE_DNS_FAIL: <407184933@qqqqq12372.com >.

A.5. Unexpected SMTP Reply Code

This example shows how MAP respond to an unexpected SMTP reply code.

S: CMD_SMTP_CONN.

SS: 554 mailprdvadmz2-mx.es.gwu.edu

S: RE_BADREPLY: <554>.

A.6. A Static IP Requirement

This example shows how MAP respond when it is rejected for a dynamic IP address.

S: CMD_SMTP_FROM.

SS: 550 5.7.1 Service unavailable, Client host [166.172.57.252] blocked using Spamhaus. To request removal from this list see http://www.spamhaus.org/lookup.lasso (AS3130). [DB5EUR03FT042.eop-EUR03.prod.protection.outlook.com]

S: RE_SERVICE_DENIED.