

MINIMUM IMPLEMENTATION OF SIMPLE MAIL TRANSFER PROTOCOL

Table of Contents

1. INTRODUCTION	1
2. THE SMTP MODEL	1
3. SMTP COMMANDS	2
4. SMTP REPLIES	3
3.1. NUMERIC ORDER LIST OF REPLY CODES	3
4.2. COMMAND-REPLY SEQUENCES	3
5. MESSAGE RETRIEVAL	4
6. MESSAGE DELETION	4
7. SECURITY CONSIDERATIONS	4
8. EXAMPLE	4
9. REFERENCES	5

1. INTRODUCTION

The purpose of this document is to explain the communication between a client host and a server host. In the next paragraphs, detailed information is provided about how a minimum implementation of SMTP model is achieved.

2. THE SMTP MODEL

The SMTP design is depended on a particular model of communication. Firstly, the sender-SMTP establishes a connection with the receiver-SMTP. Then, the sender-SMTP generates appropriate commands, which are sent to the receiver-SMTP. Finally, the receiver-SMTP, in turn, sends the appropriate replies back to the sender-SMTP.

Once the connection is established, sender sends a MAIL command specifying the sender of the mail. Whether the receiver can receive the mail, it responds with an OK reply; whether not, it responds with a declining message. Then the sender sends a RCPT command specifying the recipient of the mail. If the receiver may accept mail for that particular recipient, it responds with an OK message; if not, it responds with a declining message. Then the sender sends DATA. If the receiver can accept mail data, it responds with an OK message and sender sends the mail data. Otherwise, it responds with a refusing message.

3. THE SMTP COMMANDS

The SMTP commands are the definition of the mail system operation requested by the user.

HELO COMMAND

```
HELO <SP> <domain> <CRLF>
```

This command is used to identify the sender-SMTP to the receiver-SMTP. The argument field contains the host name of the sender-SMTP.

The receiver-SMTP makes itself known to the sender-SMTP in the connection greeting reply.

This command and an OK reply to it, declare that both the sender-SMTP and the receiver-SMTP are in the initial state.

MAIL COMMAND

```
MAIL <SP> FROM:<reverse-path> <CRLF>
```

This command initiates a mail transaction in which the mail data is delivered to the recipient. The argument to the MAIL command is a reverse-path.

Reverse-path defines who send the mail.

RCPT COMMAND

```
RCPT <SP> TO:<forward-path> <CRLF>
```

This command is used to declare the recipient of the mail. The argument to the mail command is a forward-path.

Forward-path specifies who is the recipient of the mail.

DATA COMMAND

```
DATA <CRLF>
```

This command is used to send the mail data. If receiver can accept the mail data, it responses an OK reply and then it recognizes the following lines as the mail data from the sender.

HELP COMMAND

```
HELP [<SP> <string>] <CRLF>
```

This command is used to tell the receiver to send helpful information about the commands' actions. The argument is a command name and the information that receiver sends back is for a specific command at a time.

NOOP COMMAND

```
NOOP <CRLF>
```

This command does not affect any previously entered commands or any parameters. It states no activity. It just causes the receiver to respond an OK reply.

QUIT COMMAND

QUIT <CRLF>

This command causes the receiver to send an OK reply and afterwards close the transmission channel.

4. SMTP REPLIES

Replies to SMTP commands are implemented to make sure that the requests of the sender and the actions are synchronized in the procedure of mail transfer, and to assure that the sender knows the state of the receiver every time. Every request of the sender has one reply from the receiver.

4.1. NUMERIC ORDER LIST OF REPLY CODES

220 <domain> Service ready

214 Help message

[Information on how to use the receiver or the meaning of the particular non-standard command; this reply is useful only to the human user]

221 <domain> Service closing transmission channel

250 Requested mail action okay, completed

354 Start mail input; end with <CRLF>.<CRLF>

421 <domain> Service not available, closing transmission channel

500 Syntax error, command unrecognized

504 Command parameter not implemented

4.2. COMMAND-REPLY SEQUENCES

Each command contains its possible replies. 'S' for success, 'E' error, 'F' for failure and 'I' for intermediate.

CONNECTION ESTABLISHMENT

S: 220

F: 421

HELO

S: 250

E: 500, 504, 421

MAIL

S: 250

E: 500, 504

RCPT

```

        S: 250
        E: 500, 421
    DATA
        I: 354 -> data -> S: 250
        E: 500, 421
    HELP
        S: 211, 214
        E: 500, 504, 421
    NOOP
        S: 250
        E: 500, 421
    QUIT
        S: 221
        E: 500

```

5. MESSAGE RETRIEVAL

A simple message retrieval implementation is implemented in this SMTP model. SMTP-sender(client) requests from the SMTP-receiver(server) to send back all messages(DATA) SMTP-sender sent.

MESSAGE RETRIEVAL COMMAND

```
RETR <CRLF>
```

6. MESSAGE DELETION

A simple message deletion implementation is implemented in this SMTP model. SMTP-sender requests from the SMTP-receiver to delete all messages SMTP-sender sent.

MESSAGE DELETION COMMAND

```
DELE <CRLF>
```

7. SECURITY CONSIDERATIONS

A login system is implemented for security purposes. After the connection establishment with SMTP-receiver, user must enter a correct username and password in order to login and use the program. Otherwise, if username and password combination is wrong, the connection with SMTP-receiver is terminated.

8. EXAMPLE

Example of the SMTP procedure

This SMTP example shows mail sent by mytest at host domain.gr.

```

S: MAIL FROM:<mytest@mail.gr>
R: 250 OK

S: RCPT TO:<recipienttest@domain.gr>
R: 250 OK

```

S: DATA
R: 354 Start mail input; end with <CRLF>.<CRLF>
S: Blah blah blah(MAIL CONTENT) ...
S: <CRLF>.<CRLF>
R: 250 OK

9. REFERENCES

Jonathan B. Postel, (1982). "Simple Mail Transfer Protocol". [ONLINE]
Available at: <https://tools.ietf.org/html/rfc821>.