

# Écrire un message en IMAP avec Python

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 22 septembre 2010

<https://www.bortzmeyer.org/python-imap-upload.html>

---

La bibliothèque `imaplib` <<http://docs.python.org/library/imaplib.html>> du langage Python permet d'accéder à un serveur de courrier IMAP et est surtout utilisée pour lire des messages. Mais elle peut aussi être utilisée pour écrire.

Attention, j'ai bien dit « écrire des messages dans une boîte », pas « envoyer un courrier à un destinataire », chose que IMAP ne permet pas. La méthode est simple, utilisant la méthode `append()`, et je peux mettre le code directement ici (le message est lu sur l'entrée standard mais il pourrait venir de n'importe où) :

```
connection = imaplib.IMAP4_SSL(hostname)
connection.login(username, password)
message_first = sys.stdin.readline() # Drop the mbox separator line
message = sys.stdin.read()
result = connection.append(mailbox, '', imaplib.Time2Internaldate(time.time()), \
                           message)
print result
connection.logout()
```

Et voilà, le programme, si tout s'est bien passé, mettra le message dans la boîte `mailbox`. En cas de problème, la première chose à vérifier est que le message est bien un message à la syntaxe du RFC 5322<sup>1</sup>, avec une ligne vide entre les en-têtes et le corps.

Un bon article de plusieurs exemples de programmes Python utilisant cette bibliothèque est « *imaplib - IMAP4 client library* » <<http://www.doughellmann.com/PyMOTW/imaplib/>> ».

Si, une fois qu'on a ce programme, on veut l'intégrer dans `mutt`, par exemple pour transmettre des messages de hameçonnage à la base du CRU <<https://listes.cru.fr/sympa/arc/sntp-fr/2010-09/msg00003.html>>, il suffit d'avoir le programme (en ligne sur <https://www.bortzmeyer.org/files/report-phishing.py>), un fichier de configuration comme :

---

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc5322.txt>

```
[default]
hostname: XXX.ensmp.fr
username: YYY
password: ZZZ
mailbox: INBOX.UUU
```

et de mettre dans son `/.muttrc`:

```
macro index,pager \ep "<enter-command>set pipe_decode=no\n<enter-command>set wait_key\n<pipe-entry>report-pl
```

Désormais, en regardant un message de hameçonnage, il suffira d'un ESC-p pour transmettre le message au dépôt.