imaplib模块-Python操作IMAP邮件服务器

Python的imaplib模块实现了IMAP4客户端,本帖介绍怎么使用Python操作IMAP服务器。

This module defines three classes, IMAP4, IMAP4_SSL and IMAP4_stream, which encapsulate a connection to an IMAP4 server and implement a large subset of the IMAP4rev1 client protocol as defined in RFC 2060. It is backward compatible with IMAP4 (RFC 1730) servers, but note that the STATUS command is not supported in IMAP4.

<u>imaplib</u>定义了三个类: imap4(明文传输)、impa4_SLL(ssl加密传输)、impa4_stream。它们代表impa4客户端协议的集合,本帖以IMAP4_SSL为例,其它类提供了类似的API接口。

IMAP(即Internet Message Access Protocol,互联网信息访问协议),你可以通过这种协议将邮件服务器上的邮件双向和计算机或移动设备终端同步邮件信息。IMAP4与POP3协议类似,都是邮件获取协议的一种,IMAP的最主要特点为"同步操作"。IMAP4改进了POP3的不足,用户可以设置先通过浏览邮件信件头来决定是否收取、删除和检索邮件的特定部分。

连接IMAP服务器

```
1 import imaplib
2
3 connection = imaplib.IMAP4_SSL('imap.gmail.com', 993)
4 # 使用imap, 比如要访问163邮箱,地址是imap.163.com,而不是mail.163.com
5
6 username = 'gmailusername'
7 password = 'gmailpassword'
8
9 # 登陆
10 try:
11 connection.login(username, password)
12 except Exception as err:
13 print('登陆失败::', err) # 输出登陆失败的原因
14
15 # 输出日志
```

```
16 connection.print_log()
17
18 # 断开连接
19 #connection.close()
20 connection.logout()
```

执行结果:

列出邮箱目录

```
1 res,data = connection.list()
2 print('Response code:', res)
3 print(data)
```

```
Response code: OK
[b'() "/" "INBOX"', b'(\\Drafts) "/" "&g0l6P3ux-"', b'(\\Sent) "/" "&XfJT0ZAB-"', b'(\\Trash) "/" "&XfJSIJZk-"', b'(\\Junk) "/" "&V4NXPpCuTvY-"', b'() "/" "&dcV r0mWHTvZZQQ-"', b'() "/" "&Xn9USpCuTvY-"', b'() "/" "&i6KWBZCuTvY-"', b'() "/" "Sent"', b'() "/" "Deleted Messages"', b'() "/" "Notes"']
```

和你的邮箱目录进行对比,注意有子目录的邮箱目录。

列出某个邮箱目录:

```
1 res,data = connection.list(directory='Deleted Messages') # 支持模式匹配,如
2
3 print('Response code:', res)
4 for line in data:
5 print(line)
```

邮箱信息

使用status()方法查询一些信息:

```
1 import re
2 list_response_pattern = re.compile(
3    r'\((?P<flags>.*?)\) "(?P<delimiter>.*)" (?P<name>.*)'
4 )
5
6 def parse_list_response(line):
7    match = list_response_pattern.match(line.decode('utf-8'))
8    flags, delimiter, mailbox_name = match.groups()
9    mailbox_name = mailbox_name.strip('"')
10    return (flags, delimiter, mailbox_name)
```

```
11
12 for line in data:
13    flags, delimiter, mailbox = parse_list_response(line)
14    print('Mailbox:', mailbox)
15    status = connection.status('"{}"'.format(mailbox), '(MESSAGES RECENT Uprint(status))
```

```
Mailbox: INBOX
('OK', [b'"INBOX" (MESSAGES 109 RECENT 108 UIDVALIDITY 1 UNSEEN 108)'])
Mailbox: &g0l6P3ux-
('OK', [b'"&g016P3ux-" (MESSAGES 0 RECENT 0 UIDVALIDITY 2 UNSEEN 0)'])
Mailbox: &XfJT0ZAB-
('OK', [b'"&XfJT0ZAB-" (MESSAGES 2 RECENT 0 UIDVALIDITY 3 UNSEEN 0)'])
Mailbox: &XfJSIJZk-
('OK', [b'"&XfJSIJZk-" (MESSAGES 0 RECENT 0 UIDVALIDITY 4 UNSEEN 0)'])
Mailbox: &V4NXPpCuTvY-
('OK', [b'"&V4NXPpCuTvY-" (MESSAGES 6 RECENT 6 UIDVALIDITY 5 UNSEEN 6)'])
Mailbox: &dcVr0mWHTvZZOQ-
('OK', [b'"&dcVr0mWHTvZZOQ-" (MESSAGES 0 RECENT 0 UIDVALIDITY 6 UNSEEN 0)'])
Mailbox: &Xn9USpCuTvY-
('OK', [b'"&Xn9USpCuTvY-" (MESSAGES 4 RECENT 4 UIDVALIDITY 7 UNSEEN 4)'])
Mailbox: &i6KWBZCuTvY-
('OK', [b'"&i6KWBZCuTvY-" (MESSAGES 25 RECENT 25 UIDVALIDITY 18 UNSEEN 25)'])
Mailbox: Sent
('OK', [b'"Sent" (MESSAGES 0 RECENT 0 UIDVALIDITY 257 UNSEEN 0)'])
Mailbox: Deleted Messages
('OK', [b'"Deleted Messages" (MESSAGES 0 RECENT 0 UIDVALIDITY 258 UNSEEN 0)'])
Mailbox: Notes
('OK', [b'"Notes" (MESSAGES 0 RECENT 0 UIDVALIDITY 259 UNSEEN 0)'])
```

• MESSAGES: 邮箱中消息数

• RECENT: 最近邮件数

UIDNEXT

UIDVALIDITY

UNSEEN

选择(select)邮箱目录

```
1 res, data = connection.select('INBOX')
2 print(res, data)
3 print(data[0]) # 邮件数
4 # OK [b'110']
5 # b'110'
```

我使用163邮箱测试时,输出了如下安全提示:

```
1 NO [b'SELECT The login is not safe! Please update your mail client: http://
```

如果你也使用163邮箱,网易会给你发一个确认邮件:"如您确认当前所用的客户端为可信任客户端,仍继续使用,并愿自行承担信息泄露风险和损失,可前往这里设置。",根据提示设置即可。

使用search获得消息ID:

```
1 connection.select('INBOX', readonly=True)
2 res, msg_ids = connection.search(None, 'ALL') # 你也可以直接搜索邮件
```

3 print(res, msg_ids)

OK [b'1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 5 5 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 10 6 107 108 109 110']

获得邮件

```
1 status, count = connection.select('INBOX', readonly=True)
2 res, msg_data = c.fetch(count[0], '(BODY.PEEK[HEADER] FLAGS)') # '(UID BODY print.(msg_data)
```

[(b'110 (FLAGS () BODY[HEADER] {1138}', b'Received: from edm_common_593084_31323 2 (unknown [192.168.200.104])\r\n\tby mfast9 (Coremail) with SMTP id uMCowEBpInM CRSFYW9EsEw---.425S2;\r\n\tTue, 08 Nov 2016 11:22:42 +0800 (CST)\r\nFrom: =?UTF-8 ?B?572R5piT6YKu5Lu25Lit5b+D?= <mail@service.netease.com>\r\nSender: mail@service .netease.com\r\nTo: "_____163.com" <_____163.com>\r\nMessage
-ID: <329252359.864716.1478575362502.JavaMail.mail@service.netease.com>\r\nSubje 3163.com" < ct: =?UTF-8?B?572R5piT6YKu566x5o+Q6YaS77ya6Zi75q2i5LqG?=\r\n =?UTF-8?B?5LiA5qyh5 LiN5a6J5YWo55qE5pS25L+h6K+35rGC?=\r\nMIME-Version: 1.0\r\nContent-Type: text/htm l; charset=UTF-8\r\nContent-Transfer-Encoding: quoted-printable\r\nX-Netease-Fol der: INBOX\r\nX-NETEASE-FLAG: cHJvZD1lZG0mdHlwZT1jb21tb24maWlkPTU5MzA4NCZkaWQ9Mz EzMjMy\r\nX-CM-TRANSID:uMCowEBpInMCRSFYW9EsEw--.425S2\r\nX-Coremail-Antispam: 1U f129KBjvdXoWrZr4kAr43JrWDAFW3ZryUtrb_yoWDGwb_ur\r\n\ts5JryjkF42ywsFqrZ3AF97C3yfK aykKr13XrZIgrsxCw1Fga1DX3ykAw4jyw1rtry3C3Wj\r\n\ty3s8tF45J3srGjkaLaAFLSUrUUUUUb8 apTn2vfkv8UJUUUU8Yxn0WfASr-VFAUDa7-sFnT\r\n\t9fnUUvcSsGvfC2KfnxnjZKAaVFWhVr0xZFp f9x0zRUUUUUUUUU=\r\nX-Originating-IP: [192.168.200.104]\r\nDate: Tue, 8 Nov 2016 11:22:42 +0800 (CST)\r\n\r\n'), b')']

```
1 connection.select('INBOX', readonly=True)
2 res, msg_data = connection.fetch('1', '(RFC822)')
3 for response_part in msg_data:
4    print(response_part)
```

(b'1 (RFC822 {24868}', b'Received: from mail-io0-f181.google.com (unknown [209.8 5.223.181])\r\n\tby mx24 (Coremail) with SMTP id SsCowEBJV0MQpPlXc5w9Dg--.4792S1 3;\r\n\tSun, 09 Oct 2016 10:01:13 +0800 (CST)\r\nReceived: by mail-io0-f181.goog le.com with SMTP id q192so81118321iod.0\r\n for < 163.com>; Sat, 08 Oct 2016 19:01:12 -0700 (PDT)\r\nX-Google-DKIM-Signature: v=1; a=rsa-sh a256; c=relaxed/relaxed;\r\n d=1e100.net; s=20130820;\r\n h=x-qm-m essage-state:delivered-to:dkim-signature:date:from:to:subject\r\n version:list-unsubscribe:feedback-id:message-id;\r\n bh=Fq+GYggf+1rDfbz5E fdvqSWeHVVky1I9W1WvjylQoiM=;\r\n b=VrgVHJ0s7AX8kihK76N5BdrvVfH0f6IlqfMIRl exrtVUo4vXTnSselIC0QsVj7TERE\r\n ikRTPVf5yHSmLgqGkSDg/qqxchNMX47015+tBE2 uNFBRCRC3N/yNarMGjHUgPy5DxFTK\r\n Dh3kpxv5vbWQM++J/Zrxthg1zBtmEvVa2uxaHD ZRDHt0k3vhshVZH0KICSDpNs9I8ruV\r\n RKic0AvHl+dgIvFH0n8ONjXZ9sftYrDCleg4F XtYVdBsgbmFjn07CwNUuDbDtbGiawub\r\n +r2S9DYWo0pXsu0FEkAwj5eV/Czhjy7yjjbP ACU8kXGFrw8ESoHLH5cU1gSuXrqpxvwq\r\n Gvxw==\r\nX-Gm-Message-State: AA6/9 RktE5/PmVKqk@um2hnnadfhmqg5PHxBa1axZCSxJQOvzxTkDL/H5aUoCt3Gy3zFpg6N@Z9CK8@YfSYJ2 AQjxwAM5sU=\r\nX-Received: by 10.107.138.204 with SMTP id c73mr31612428ioj.192.1 Sat, 08 Oct 2016 19:01:12 -0700 (PDT)\r\nX-Forwarded-To 475978472504;\r\n 163.com\r\nX-Forwarded-For: ____3gmail.com 009@163.com\r\nDelivered-To: igmail.com\r\nReceived: by 10.36.28.21 2 with SMTP id c203csp974708itc;\r\n Sat, 8 Oct 2016 19:01:11 -0700 (PDT) \r\nX-Received: by 10.98.87.69 with SMTP id l66mr40094090pfb.128.1475978471406;\ WIF Daily Blug

上面邮禅是gmail转发给163的html格式邮件,比较乱。

```
connection.select('INBOX')
2
3 # 邮件ID
4 res, [msg_ids] = connection.search(None, 'ALL')
5 print(msg_ids)
6
7 # 查找邮件
8 res, [msg_ids] = connection.search(None,'(SUBJECT "Hello World")',)
9 msg_ids = ','.join(msg_ids.decode('utf-8').split(' '))
10 print(msq_ids)
11
12 # flags
13 res, response = connection.fetch(msg_ids, '(FLAGS)')
14 print(response)
15
16 # 标记为Deleted flag
17 res, response = connection.store(msg_ids, '+FLAGS', r'(\Deleted)')
18
19 # 查看flag
20 res, response = connection.fetch(msg_ids, '(FLAGS)')
21 print(response)
22
23
24 # 查找邮件
25 res, [msg_ids] = connection.search(None, '(SUBJECT "Hello World1")',)
26 msg_ids = ','.join(msg_ids.decode('utf-8').split(' '))
27 print(msq_ids)
28
29 # 删除邮件
30 res, response = connection.expunge()
31 print(response)
32
33 # 创建邮箱目录
34 res, create_response = connection.create('Notes.Good')
35 print(create_response)
36
37 # 复制邮件到新目录
38 print(msg_ids)
39 connection.copy(msq_ids, 'Notes.Good')
41 # 移动用move()方法
```

在邮箱中添加新邮件:

```
1 new_message = email.message.Message()
2 new_message.set_unixfrom('test')
3 new_message['Subject'] = 'subject'
4 new_message['From'] = 'test1@example.com'
5 new_message['To'] = 'test2@example.com'
6 new_message.set_payload('This is the body of the message.\n')
7
8 connection.append('INBOX', '',imaplib.Time2Internaldate(time.time()), str(r
```

• imaplib文档: https://docs.python.org/3/library/imaplib.html

f Facebook S⁺ Google+ **y** Twitter **♂** Weibo **⋈** Email

相关文章

Python中判断一个文件是否存在
查看python已经安装的模块
计算在美国进行公路旅行的最优路线
用Python获取本地IP地址
Python 的多重继承
iii 2016年10月30日 💄 EatPython 🕒 coding 🕜 imaplib、python
《imaplib模块-Python操作IMAP邮件服务器》有1个想法
.

kylin

2017年1月12日 上午10:19

在使用search(None,'(SUBJECT "XXXX")'的时候,网易邮箱总是返的空。 邮箱里面是有该主 题的邮件的。

十分困惑。

Copyright © 2013-2017 WTF Daily Blog | Powered by DigitalOcean