Filtriranje duplikata mailova putem skripte cleanupmbox.py

Sigurno vam se već dogodilo da u sandučiću elektroničke pošte nađete duplikate poruka. Razlog može biti pogreška POP/IMAP poslužitelja, pogreška vašeg mail klijenta ili jednostavno ljudska pogreška kod konfiguracije (višestruki korisnički računi i slično). Problem će nam pomoći riješiti mala skripta, pisana u programskom jeziku Python.

Skripta će analizirati vaš sandučić, te po zaglavlju Message-ID kojeg ima svaka poruka, izbaciti duplikate. Ovi duplikati su filtrirani isključivo po Message-ID zaglavlju, poruke koji imaju samo istog pošiljatelja i isti Subject neće biti smatrani duplikatima.

Zaglavlje Message-ID izgleda kao dugačak niz znakova, i jedinstveno označava svaku poruku:

```
Message-Id: <1291194791.279685.342236381.5436.1@server.carnet.hr>
```

Po ovom zaglavlju se može prepoznati da je poruka identična nekoj drugoj, a neki mail sustavi čak automatski sprječavaju isporuku identičnih poruka. Drugi sustavi to mogu, ali tek uz neki dodatak (plugin). Postoje dodaci i za mail klijente, primjerice Evolution. No, sve je to kasno ukoliko imamo sandučić sa tisućama mailova u kojem biste htjeli maknuti višak.

Ovdje u pomoć priskače ova jednostavna skripta, koja se pokreće na ovaj način:

```
cleanupmbox.py -i Moj_mailbox -o Moj_mailbox.ok -h mbox.h
```

Moj_mailbox je sandučić kojeg želite pregledati, i ne smije biti u uporabi (kako se sadržaj ne bi promijenio tijekom analize). U suprotnom, riskirate da izlazni sandučić bude nečitljiv.

S opcijom -i određujete ulaznu, a s opcijom -o određujete izlaznu datoteku.

Opcija -h će generirati indeks (hash) datoteku, kako biste kasnije mogli brže napraviti redukciju sandučića, ali nije ju neophodno rabiti, pogotovo ako sandučić nije prevelik.

Primjer uporabe:

```
$ ./cleanupmbox.py -i Moj_mailbox -o Moj_mailbox.ok
New Message-ID: <240427k2f81ed7ah77e493759eec4b6e@mail.gmail.com>
New Message-ID: <A4E927A395F9898133A655D43621D@hpml350.Imun1.local>
New Message-ID: <48DB7261.2068734@domena.hr>
Duplicate Message-ID: <48DB7261.2068734@domena.hr>
New Message-ID: <48E26934.7030004@ptfos.hr>
New Message-ID: <rt-3.6.1-16731-1287834558-1772.3338-8-0@carnet.hr>
...
```

Program će za svaki novi Message-ID ispisati "New Message-ID", a kada nađe duplikat ispisat će, naravno, "Duplicate Message-ID". U datoteci Moj_mailbox.ok će biti skraćena inačica sandučića, bez duplikata.

Copyright ©2003.-2017. CARNet. Sva prava pridržana.
Mail to portal-team(at)CARNet.hr

Preporučujemo da skriptu primjenite ukoliko arhivirate poštu nekog neaktivnog korisnika, ili na velikim sandučićima gdje se uvijek mogu naći duplikati. Na taj način možete uštediti nešto prostora na disku i drugim resursima, što na velikom broju korisnika može biti primjetno.

Autor skripte je Marilen Corciovei (len@len.ro), a originalnu skriptu možete skinuti ovdje [1]. Ukoliko se pojavi nova inačica, možete ju pronaći na njegovom blogu na adresi: http://www.len.ro/2009/01/remove-duplicate-mails/ [2]

```
#!/usr/bin/env python
# author Marilen Corciovei len@len.ro, this code is offered AS IS, use at
# your own risk

import re, sys, email, getopt, marshal

msg_start = 'From'
cleaned = None
mids = {}

def parse_mbox(file_name):
    file = open(file_name, 'r')
    msg = ''
    lastLine = ''
    while 1:
        line = file.readline()
        if not line: break
        if line.startswith(msg_start) and lastLine == '':
            if len(msg) > 0:
                parse_msg(msg)
            msg = ''
        msg = msg + line #+ '\n'
        lastLine = line.strip()

def parse_msg(smsg):
    m = email.message_from_string(smsg)
    if 'message-id' in m:
        mid = m['message-id']
        if mid in mids:
            print 'Duplicate Message-ID:', mid
        else:
            print 'New Message-ID:', mid
            mids[mid]=mid
            cleaned.write(smsg)

if __name__=='__main__':
    in_file = ''
    out_file = ''
    hash_file = ''
    try:
        opts, args = getopt.getopt(sys.argv[1:], "i:o:h:")
    except getopt.GetoptError:
        print 'Usage', sys.argv[0], '-i input -o output [-h hash file]'
        sys.exit(2)
    for o, a in opts:
        if o == "-i":
            in_file = a
        if o == "-o":
            out_file = a
```


Copyright ©2003.-2017. CARNet. Sva prava pridržana.
Mail to portal-team(at)CARNet.hr

```
        if o == "-h":
            hash_file = a

    if in_file == '' or out_file == '':
        print 'Usage', sys.argv[0], '-i input -o output [-h hash file]'
        sys.exit(2)

    #global cleaned
    cleaned = open(out_file, 'w')
    if hash_file != '':
        try:
            mids = marshal.load(open(hash_file,'r'))
        except:
            pass

    parse_mbox(in_file)
    if hash_file != '':
        marshal.dump(mids, open(hash_file,'w'))
```

Prilog	Veličina
cleanupmbox.py [1]	1.67 KB

- Logirajte [3] se za dodavanje komentara

sri, 2010-12-01 13:40 - Željko Borš**Kuharice:** Linux [4]
Kategorije: Software [5]
Servisi [6]
Vote: 0

No votes yet

Source URL: https://sysportal.carnet.hr/node/799

Links
[1] https://sysportal.carnet.hr/system/files/cleanupmbox.py_.txt
[2] http://www.len.ro/2009/01/remove-duplicate-mails/
[3] https://sysportal.carnet.hr/sysportallogin
[4] https://sysportal.carnet.hr/taxonomy/term/17
[5] https://sysportal.carnet.hr/taxonomy/term/25
[6] https://sysportal.carnet.hr/taxonomy/term/28

Copyright ©2003.-2017. CARNet. Sva prava pridržana.
Mail to portal-team(at)CARNet.hr