

## Laboratorul 11 - E-mail

Responsabili: Radu-Ioan Ciobanu, Bianca Oprea

### Obiective

În urma parcurgerii acestui laborator, studentul va fi capabil să:

- diferențieze și utilizeze două protocoale pentru citirea poștei electronice
- folosească protocolul pentru trimiterea de mesaje și atașamente prin poșta electronică
- scrie un client simplu de e-mail.




### Introducere

Primul mesaj e-mail a fost transmis în 1971 de un inginer pe nume Ray Tomlinson. Până la acea dată, puteau fi trimise mesaje doar în cadrul aceleiași calculator. Marea îmbunătățire introdusă de Tomlinson a fost posibilitatea de a trimite mesaje între calculatoare diferite din Internet, folosind semnul '@' pentru a desemna mașina spre care se trimite mesajul.

Azi se trimit miliarde de mesaje e-mail pe zi, si totuși multe din caracteristicile de atunci ale mesajelor au rămas. Vom vedea în continuare structura mesajelor e-mail și modul în care sunt transmise acestea în Internet.

### Structura mesajelor e-mail

Un mesaj e-mail a fost întotdeauna transmis în format plain-text (text clar). Chiar si prin adăugarea atașamentelor, mesajele de e-mail sunt trimise tot ca mesaje plain-text, prin folosirea unor mecanisme de codificare (*uuencode/uudecode*, *MIME/BASE64*).

Un mesaj este format dintr-o secțiune de antete (headers), urmată de o secțiune cu conținutul mesajului. Structura antetelor este descrisă în  RFC 822,  RFC 1521 și  RFC 1806, ele având în general următoarea structură:

- unul sau mai multe antete *Received*, care indică ce cale a fost urmată de mesaj de la sursa până la destinație
- Mime-Version*: versiunea MIME (Multipurpose Internet Mail Extensions) folosită, 1.0 în general
- Content-Type*: *text/plain* pentru mesaje text, *multipart/mixed* pentru mesaje cu atașamente
- Subject*: subiectul mesajului
- Date*: data și ora când a fost trimis mesajul
- Message ID*: un ID pentru mesaj, folosit pentru identificarea în mod unic a unui mesaj
- From*: numele și adresa de mail a expeditorului
- To*: numele și adresa de mail a destinatarului
- Cc*: carbon copy (alți destinatari)
- alte antete introduse de clientul de e-mail folosit pentru a trimite mesajul.

Conținutul mesajului este textul propriu-zis, pentru mesajele în text clar fără atașamente. Se poate observa mai jos un exemplu de mesaj:

```
MIME-Version: 1.0
From: profesor@upb.ro
To: student@upb.ro
Subject: Tema
Content-Type: text/plain
```

Draga student,

Fa-ti tema!

Cu bine,  
Profesorul.

Mesajele cu atașamente pot folosi una din următoarele tehnici pentru codificarea acestora:

- uuencode* - la începuturile e-mail-ului, fișierele care se doreau trimise trebuiau convertite în format text și invers prin folosirea utilitărelor numite *uuencode/uudecode*; și în ziua de azi, unii clienți de mail adaugă atașamentele la sfârșitul mesajelor, codificându-le cu algoritmul folosit de *uuencode*
- MIME/Base64* - această tehnologie este cea recomandată pentru trimiterea de mesaje cu atașamente.

Un mesaj cu atașamente codificate  MIME arată în felul următor:

```
MIME-Version: 1.0
From: Student Studentescu <student@upb.ro>
To: Profesor PC <profesor@upb.ro>
Subject: Re: Tema
Content-Type: multipart/mixed; boundary=abc
```


```
--abc
Content-Type: text/plain
```

Atasez tema.

Cu bine,  
Studentul

```
--abc
Content-Type: text/plain
Content-Disposition: attachment; filename="tema.c"
```

```
#include <stdio.h>
int main()
{
    printf("Aceasta este tema mea\n");
    return 0;
}
--abc
```



Se observă faptul că părțile care compun mesajul sunt separate între ele printr-un șir de caractere separator (boundary string), specificat ca un parametru pentru antetul *Content-Type*. Fiecare parte poate avea la rândul ei propriile antete, care conțin în general tipul și numele fișierului din secțiunea respectivă. În cazul în care se trimit atașamente binare, acestea sunt codificate folosind schema numită Base64, descrisă în  RFC 1521.

### Protocoalele SMTP, POP3 și IMAP

În terminologia folosită de sistemele de e-mail, există trei actori. Aceștia pot fi situați pe trei mașini diferite sau pot co-exista pe aceeași gazdă:

- Mail User Agent (MUA) - aplicația folosită de utilizator pentru a citi și trimite mesaje e-mail (clientul de e-mail); el nu primește direct mesaje, acesta fiind rolul Mailbox Server-ului
- Mailbox Server - serverul care primește și stochează mesajele (server de e-mail)
- Mail Transfer Agent (MTA) - aplicația care primește și retrimite mesajele spre un alt MTA sau spre un Mailbox Server ("router" de e-mail).

#### SMTP

SMTP (Simple Mail Transfer Protocol) este un protocol care se folosește pentru trimiterea mesajelor electronice (de la un client către un server). Acesta se foloseste de portul 25 peste TCP și este descris în  RFC 821 și  RFC 5321.

Mesajele necesare în SMTP pentru trimiterea unui e-mail sunt următoarele:

```
1: HELO client.upb.ro
2: MAIL FROM: <profesor@upb.ro>
3: RCPT TO: <student@upb.ro>
4: DATA
5: MIME-Version: 1.0
From: profesor@upb.ro
To: student@upb.ro
Subject: Tema
Content-Type: text/plain
```

Draga student,

Fa-ti tema!

Cu bine,  
Profesorul.

.

```
6: QUIT
```

Se trimite deci întâi o comandă "HELO" cu numele de domeniu sau adresa IP a clientului pentru a iniția sesiunea, apoi o comandă "MAIL FROM" cu adresa sursei, "RCPT TO" pentru destinație, "DATA" pentru date (e-mail-ul în sine) și "QUIT" pentru a se închide sesiunea. Secțiunea de date trebuie neapărat terminată cu secvența de caractere <CR><LF>.<CR><LF> (adică o linie nouă urmată de un punct și apoi de încă o linie nouă).

#### POP3

POP3 (Post Office Protocol 3) este un protocol utilizat pentru citirea mesajelor electronice (de la un server către un client). Clientul va interoga periodic serverul, va descărca mesajele și le va șterge automat de pe server. Comunicația se realizează folosind portul 110 peste TCP, în felul următor:

```
1: USERNAME username
2: PASS password
3: LIST
4: RETR 1
5: QUIT
```

#### IMAP

IMAP (Internet Message Access Protocol) este un protocol care se folosește pentru citirea mesajelor electronice (de la un server catre un client). Clientul interoghează periodic serverul și poate cere mesaje complete sau doar porțiuni (header, body), și nu va șterge automat mesajele de pe server. Comunicația se realizează prin TCP, folosind portul 143.

```
1: LOGIN username password
2: LIST "" "*"
3: EXAMINE Inbox
4: FETCH 1 BODY[]
5: LOGOUT
```



### Exerciții

În acest laborator, veți folosi protocolul SMTP pentru a trimite e-mail-uri către un server SMTP ce rulează local. Aceasta poate fi simulat folosind un utilitar existent în Python, numit *smtpd*. Rularea acestui utilitar pe portul 25 se face astfel:

```
python -m smtpd -n -c DebuggingServer 127.0.0.1:25
```



Atenție! Pentru a putea rula serverul *smtpd* pe un port mai mic de 1024, trebuie să aveți drepturi de *sudo*.


- După ce ați pornit serverul *smtpd* într-un terminal, folosiți utilitarul *telnet* într-un terminal separat pentru a trimite un e-mail către server. *telnet* se pornește în felul următor: `telnet 127.0.0.1 25`. În acest moment, s-a deschis o conexiune TCP pe portul dat ca parametru, iar comunicația SMTP se realizează prin trimiterea comenzilor specificate mai sus (HELO, MAIL FROM, etc.). Trimiteți un e-mail către server care să conțină cel puțin antetele *MIME-Version*, *From*, *To*, *Subject*, *Content-Type*.
- Pornind de la  acest schelet de cod, implementați un client SMTP peste TCP prin care să trimiteți către serverul *smtpd* un e-mail care conține niște text și un fișier dat ca parametru sub forma unui atașament de tip *text/plain* ([API](#)-ul necesar pentru conexiunea TCP cu server-ul este detaliat în  laboratorul 7).



Atenție! Pentru serverele de mail care folosesc conexiuni SSL (Secure Sockets Layer), comanda de *telnet* trebuie înlocuită cu:

```
openssl s_client -connect <ADRESA_SERVER>:<PORT_SERVER> -quiet
```

### Bonus

- Folosind instrucțiunile de  aici, trimiteți un e-mail către asistent prin intermediul serverului SMTP de la Google.

#### Cursuri

- Cursul 01.
- Cursul 02.
- Cursul 03.
- Cursul 04.
- Cursul 05.
- Cursul 06.
- Cursul 07.
- Cursul 08.
- Cursul 09.
- Cursul 10.
- Cursul 11.
- Cursul 12.

#### Laboratoare

- Laboratorul 01 - Notiuni pregatitoare pentru laboratorul de PC
- Laboratorul 02 - Folosirea unei legaturi de date pentru transmiterea unui fisier
- Laboratorul 03 - Implementarea unui protocol cu fereastra glisanta. Suma de control
- Laboratorul 04 - Forwarding
- Laboratorul 05 - ICMP
- Laboratorul 06 - Socketi UDP
- Laboratorul 07 - Protocolul de transport TCP
- Laboratorul 08 - TCP și multiplexare I/O
- Laboratorul 09 - Protocolul DNS
- Laboratorul 10 - Protocolul HTTP
- Laboratorul 11 - E-mail
- Laboratorul 12 - Protocoale de securitate. OpenSSL CLI tools
- Laboratorul 13 - Protocoale de securitate. utilizarea programatica

#### Resurse

- Mașina virtuală

#### Table of Contents

- Laboratorul 11 - E-mail
  - Obiective
  - Introducere
  - Structura mesajelor e-mail
  - Protocoalele SMTP, POP3 și IMAP
    - SMTP
    - POP3
    - IMAP
  - Exerciții
  - Bonus