

## PHP PEAR (2. rész)

Ezúttal a PHP PEAR levélkezelő osztályaival ismerkedünk meg, amelyekkel egyszerűvé válik a csatolt fájlokat és színes képeket, szövegeket tartalmazó üzenetek kezelése.

**S**orozatunk előző részében – mely az adatbázisok egységesített kezeléséről szólt – már találkoztunk a PEAR-rel. Arról azonban nem beszéltem, hogy a PEAR adatbáziskezelési képességein kívül még megannyi hasznos szolgáltatást tartogat a számunkra. Számos mindennapos feladatra – legyen szó akár adatbázis- vagy hibakezelésről, hálózatokról, akár XML-ről – a PEAR kész megoldásokkal szolgál. Ugye a `mail()` függvényt te is sokszor használtad már? Valószínűleg. Bizonyára azt is tapasztaltad, hogy ezt a függvényt nem éppen arra tervezték, hogy tetszetős HTML-leveleket küldjünk vele. Sőt, egy egyszerű csatolt fájl elküldése is lehetetlen vállalkozásnak tűnhet a megfelelő segédeszközök nélkül. Ilyenkor lehet hasznunkra PEAR MIME levélkezelésre szolgáló bővítménye.

### A MIME

A MIME a Multipurpose Internet Mail Extensions rövidítése. A MIME révén válik lehetővé – nemcsak PHP-ből, hanem bármely szabványos levelezőügyfélből –, hogy csatolt fájlokat, vagy több részből álló üzeneteket küldjünk. De a MIME felelős a levelek tartalmának helyes megjelöléséért is, és nélküle még egy ékezetes levél megírása is nehézségekbe ütközne. A MIME a szabványos RFC 822-es elektronikus levél kiterjesztéseként jött létre. Az RFC 822 maga a szabvány, ebben van meghatározva mindaz, amit ma az Interneten a levél fogalmával azonosítunk. A MIME pedig egy bővítmény a szabványhoz, amelynek ugyanúgy megvannak a maga RFC-i.

### Az első lépés

Mielőtt bárminek nekikezdenénk, telepítsük a PEAR levélkezelésre szakosodott Mail-bővítményét a következőképpen:

```
# pear install http://pear.php.net/get/Mail
```

Ahhoz, hogy ez a parancs sikeresen lefusson, először a PEAR telepítésére lesz szükség, ahogyan ezt az előző cikkben már taglaltam. Ha elakadnál, lapozd fel a Linuxvilág előző számát vagy látogass el a <http://pear.php.net/> oldalra, ahol minden tudnivalót megtalálsz.

Ezt követően pedig telepítsük a `Mail_mime`, `Mail_mimePart` és `Mail_RFC822` bővítményeket is.

### Egyszerű levélküldés

Ha nem vagyunk nagy igényűek, és csupán egy egyszerű, ékezetektől mentes levelet szeretnénk küldeni, a `Mail` osztályt is használhatjuk, ami a PHP `mail()` függvényétől talán csak annyiban különbözik, hogy a parancsfájlból meghatározhatunk bizonyos beállításokat. Például megadhatjuk, hogy a program a `Sendmail` használja a levél elküldésére vagy – mint ahogyan az *1. listán* szereplő példánkban is kitűnik – egy külső SMTP-kiszolgálót használjunk erre a célra. Mielőtt kipróbálnánk a programot, ne felejtjük el telepíteni a `PEAR_Net_Socket` és `Net_SMTP` bővítményeit.

A példában látható kód viszonylag egyszerű: a `$params['host']` változóban adjuk meg a küldéshez használni kívánt SMTP-kiszolgálót (ami jó esetben *mail.szolgáltatonkneve.hu* formájú), a `$headers` tömbben pedig a levél fejlécének elemeit határozzuk meg. A `$headers['To']` és a `$recipients` változó annyiban különbözik, hogy a `$recipients` változóban a levél tényleges címzettjeit kell megadni, míg a `$headers['To']`-ban csak a levél fejlécében látható címzett nevét. Ez utóbbi tartalma a levél küldése szempontjából közömbös, ennek ellenére ezt sem árt helyesen kitölteni.

A `Mail::factory()` tagfüggvény egy `mail` objektummal tér vissza, amelyet a `send()` tagfüggvénnyel bocsáthatunk útjára.

1. lista Egyszerű levélküldés az `smtpsend.php` segítségével

```
<?php
include('Mail.php');
$recipients = 'ggabor@sopron.hu';
$headers['From'] = 'valaki@example.com';
$headers['To'] = 'ggabor@sopron.hu';
$headers['Subject'] = 'Teszt üzenet';
$body = 'Ez egy probalevel.';
$params['host'] = 'mail.sopron.hu';

// A Mail osztály factory() tagf ggvnyövel
lötrehozunk egy mail objektumot
$email =& Mail::factory('smtp', $params);
$email->send($recipients, $headers, $body);
?>
```

Az ilyen módon küldött levelek semmiképpen nem kódolódnak, egyszerű 7 vagy 8 bites levélként postázódnak, ezért ha a levélbe ékezetet is írunk, fennállhat a veszély, hogy a címzett esetleg képtelen lesz elolvasni a neki szánt üzenetet. Emiatt csak akkor küldjünk levelet ilyen módon, ha biztosan nem használunk ékezeteket.

### A színefalak mögött

Ahhoz, hogy MIME-leveleket küldjünk, nem árt, ha egy kicsit belepillantunk, mit is takar ez a négy betű valójában. Erre azért van szükség, hogy később, amikor MIME-leveleket állítunk össze, megértsük az eljárás részleteit. Nem kell megjegyezni, nem lesz ez olyan bonyolult. Nem is szaporítom tovább a szót, nézzük!

Minden MIME-levél fejléce tartalmaz bizonyos elemeket, amelyek a levelet megjelenítő programnak szólnak. A MIME a fejléceit nemcsak a szabványos üzenet fejlécéhez adhatja hozzá, hanem a saját részüzeneteihez is. Részüzenet alatt a többrészes – vagyis a HTML-szöveget vagy mellékletet tartalmazó – üzeneteknél a levélben található

## 2. lista Példa MIME-levéltre

```
From: "Gludovatz Gabor" <ggabor@sopron.hu>
To: Valaki <valaki@example.com>
Subject: Egy egyszeru MIME level
Date: Wed, 20 Jun 1999 17:18:47 +0100
Message-ID: <000000123@localhost>
MIME-Version: 1.0
Content-Type: multipart/mixed;
boundary="Linuxvilag-123";
Content-Transfer-Encoding: 7bit
```

Ha az uzenetnek ez a resze lathato, akkor a levelet olyan levelezoprogramban olvassak, mely nem kezeli a MIME uzeneteket.

```
Linuxvilag-1233
Content-Type: text/plain; charset="iso-8859-2"
Content-Transfer-Encoding: quoted-printable
```

Hell=F3 Valaki! Ez itt egy MIME e-mail!

Gabor

```
Linuxvilag-1233
Content-Type: image/jpeg; name="kep.jpg";
Content-Transfer-Encoding: base64
Content-Disposition: attachment
<a kep.jpg tartalma base64-gyel k dolva>
```

Linuxvilag-123 --

HTML-részt vagy fájl mellékletet értjük.

A MIME fejlécek többsége mind az elsődleges fejlécben, mind a részüzenet fejlécében megjelenhet, kivétel ez alól a **MIME-Version**: mező, mely az adott levél létrehozásához használt MIME-változatot tartalmazza.

A **Content-Type**: mező az üzenet tartalmára utal. Lehetséges értékei például *text/plain*, *text/html*, *image/jpeg*, amelyek a levél vagy a részüzenet tartalmára utalnak. Ha a **Content-Type**:-ot az elsődleges fejlécben találjuk meg, az értéke valamilyen *multipart/\** érték is lehet, például *multipart/mixed* vagy *multipart/alternative*.

A **Content-Transfer-Encoding**: az adott üzenetrészben található tartalom kódolására utal. Mivel alapértelmezés szerint a levelekben csak 7-bites adatot küldhetünk, olyan módszereket kellett kitalálni, amelyekkel lehetővé válik 8-bites ékezetes karakterek, illetve fájlok küldése. Ékezetes tartalom küldéséhez a „quoted-printable” kódolást szokás használni, ha azonban a levél nem tartalmaz ékezetes karaktereket, elegendő a 7 bit is. Bináris fájlok esetén a base64 kódolás használatos, ami a fájlt úgy alakítja át 7-bitesre, hogy az később kibontható, felhasználható legyen. A 8 bit és binary kódolási típusok használata kevésbé elterjedt (emiat a működésük sem mindenütt biztosított).

A **Content-Disposition**: egy egyelőre még kísérleti állapotban lévő fejléc. Segítségével a levélolvasó program egyszerűen eldöntheti, hogy mellékletről vagy a levél tartalmához kapcsolódó anyagról van-e szó.

MIME-levéltre láthatunk példát 2. listánkon.

## 3. lista A mimemail.php – HTML-t is tartalmazó üzenet küldése

```
<?php
include('Mail.php');
include('Mail/mime.php');

$text = 'A levöl sz veges v&lt;tozata.';
$html = '<html><body>A levöl <b>HTML</b>-es
        v&lt;tozata.</body></html>';

$file = '/tmp/logo.jpg';
$crlf = "\r\n";
$headers = array(
    'From' => 'ggabor@sopron.hu',
    'Subject' => 'Pr ba zenet'
);

$mime = new Mail_mime($crlf);

$mime->setTXTBody($text);
$mime->setHTMLBody($html);
$mime->addAttachment($file, 'image/jpeg');

$body = $mime->get();
$headers = $mime->headers($headers);

$params["host"] = "mail.sopron.hu";
$mail =& Mail::factory('smtp', $params);
$mail->send('ggabor@sopron.hu', $headers,
    $body);
?>
```

## 4. lista Az rfc822.php levél cím ellenőrzése

```
<?php
$cim = 'Egy csoport: "Gludov&lt;tz G&lt;bor"
<ggabor@sopron.hu>;, valaki@example.com
(Egy megjegyzo)s';
$rfc822 = new Mail_RFC822($address,
    'example.com', true);

$cimek = $rfc822->parseAddressList();
print_r($cimek);
?>
```

## HTML-üzenetek küldése

Amennyiben levelünket HTML-üzenetként szeretnénk elküldeni, a Mail\_mime osztályhoz fordulhatunk segítségért. Az ezzel a bővítménnyel létrehozott üzenetekben található ékezetes és fájlok a szabványnak megfelelően kódolódnak, így nyugodtan bármilyen szöveget írhatunk, biztosak lehetünk benne, hogy a címzett el fogja tudni olvasni a levelet – már amennyiben a levelezőprogramja támogatja a MIME-t (a ma használatos levelezőprogramok szinte kivétel nélkül támogatják). Programunk forráskódja a 3. listában olvasható. A levél közvetlen elküldéséhez ezúttal is a PEAR Mailt hívtuk segítségül, a levél tartalmát azonban a Mail\_mime osztállyal állítjuk elő. Az osztály használata meglehetősen egyszerű,

beállítjuk levelünk szöveges tartalmát, beállítjuk a HTML-tartalmat, csatolunk egy képet, és már küldhetjük is a levelet, anélkül, hogy bele kellene bonyolódnunk a részletekbe. HTML-ben írt levél esetében a levélhez mellékelni kell az üzenet egyszerű szöveges példányát is, arra az esetre, ha a címzett levelezőprogramja nem képes html-es üzenetek megjelenítésére. A szöveges és html-es tartalom közül a címzett levelezőprogramja azt a részt fogja kiválasztani, amelyek számára a legmegfelelőbb.

Az így létrejött levél *multipart/mixed* lesz, amelynek második részüzenete *image/jpeg* típusú, ez tartalmazza a mellékelt képet; ezzel szemben első részüzenete ugyancsak *multipart* típusú, mégpedig *multipart/alternative*.

A *multipart/alternative* rész egy beágyazott MIME-üzenetet takar további részüzenetekkel, de a *multipart/mixed*-hez képest a különbséggel, hogy a *multipart/alternative* részben mindegyik melléklet ugyanazt a tartalmat hordozza, és a levelezőprogram a *multipart/alternative*-ből választja ki azt a részt, ami számára a legmegfelelőbb. Ez annyit tesz, hogyha egy levelezőügyfél nem tud HTML-t, a *text/plain* típusút választja; ezzel szemben egy mindenféle extrával felszerelt levelezőprogram a *text/html* típust fogja kiválasztani.

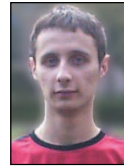
### Levélcímek ellenőrzése

A PEAR levelekkel kapcsolatos fejezetében található még egy hasznos osztály, amivel az oldalainkon megadott levélcímeket ellenőrizhetjük, hogy valóban szabályosan adták-e meg őket. Ez az osztály azért hasznos, mert egy levélcím ellenőrzése sokszor nagyon bonyolult feladat, és ha nem akarunk hamisnak tűnő, ám mégis szabályos címeket is eldobni, ezt az osztályt ajánlatos

segítségül hívni. Az így ellenőrzött levélcímek biztosan megfelelnek az RFC 822-ben meghatározott szabványnak. A bővítmény használatára a 4. listában látható egy példa.

### Összegzés

A levélküldés bizony néha bonyolultabb, mint elsőre feltételeztük volna. Ilyenkor jönnek jól a PEAR levélküldésre szakosodott osztályai, amelyekkel bármilyen típusú levelet könnyedén elküldhetünk. A PEAR tehát ezúttal is jó segítő társnak bizonyult.



Gludovátz Gábor

(ggabor@sopron.hu) egy soproni cég Linux-rendszerekkel foglalkozó rendszergazdája.

Kedvenc időtöltése a programozás, és a Linux lelkevilágában való kutakodás. Ha ideje engedi, szívesen hódol szenvedélyének és

bringáján a környező erdőket járja. Honlapja a

☞ <http://www.sopron.hu/~ggabor/> címen érhető el.

### Kapcsolódó címek

PEAR ☞ <http://pear.php.net/>

PEAR-leírás ☞ <http://pear.php.net/manual/en/>

A MIME működése

☞ <http://www.zend.com/zendspotlight/sendmimepart1.php>

☞ <http://www.phpbuilder.com/columns/Okartic20000807.php3>

RFC-k keresése ☞ <http://rfc.sunsite.dk/>

A Linux nem csak operációs rendszer: egyesek szerint életézés, mások egyenesen valóságként élik meg. Az igazi Linux-felhasználó szabadnak érezheti magát: a szabad program továbbadható másoknak, korlátlan számban másolható és felhasználható. Kezünket nem kötik ipari titokként féltve őrzött kódok, és a hozzáférésen túl a programok módosítása is lehetséges, ami azt is jelenti, hogy előttünk százak és ezrek már éltek e lehetőséggel, kijavítva számos hibát. A Linux a legtöbb élők munkát tartalmazó operációs rendszer, emellett terhelhető, igen megbízhatóan és hatékonyan használja a számítógép nyújtotta erőforrásokat – és nem utolsósorban használata komoly költségmegtakarítást jelent. *Pere László* kötete ugyanakkor nem hallgat kedvenc rendszerünk hibáiról, hiányosságairól sem. Természetesen hiteltelenné válik az a könyv, amelyben nincs lelkesedés, így a „térítésnek” is fenntart egy kicsiny fejezetet, a könyv többi része azonban a száraz tényeket mutatja be, körüljárva minden területet, amely a kezdő és haladó Linux-felhasználó számára fontos lehet.



ISBN: 963 9301 37 x  
 Ár: 2660 Ft, 249 oldal  
 Felhasználói szint: kezdő

A főbb témakörök:

- A Linux felépítése
- Bejelentkezés a rendszerbe, a felhasználó azonosítása

- A konzol használata
- A könyvtárbejegyzések
- A héj használata: parancsok, utasításcsovek, a héj testreszabása és programozása
- Állománykezelés: hivatkozások, meghajtók beillesztése a rendszerbe, tömörítés, biztonsági mentések
- Szabályos kifejezések és szűrők
- Feladatvezérlés: a folyamatok irányítása és figyelése
- Kapcsolattartás és levelezés
- Szövegszerkesztés: a vi és a Midnight Commander használata, külön táblázatokkal, amelyek a billentyűparancsokat tartalmazzák
- Héjprogramozás
- Számítógép-hálózatok és a Világháló
- Grafikus felületek: az X Window rendszer és az ablakkezelők
- Kiadványszerkesztés: részletes LaTeX-ismertető, az alapvető dokumentumformázási parancsoktól a különleges karakterek és matematikai képletek használatáig, illetve az utómunkálatokig, részletes táblázatokkal.