© Kiskapu Kft. Minden jog fenntartva

A konzolos levelezés alapjai és a titkosítás

Ha utánagondolunk, egy levél elküldése során legalább két gépen halad keresztül, és felmerülhet bennünk a gyanú, hogy megőrzi-e hitelességét és épségét.

Nem új igény ez az emberiség számára, már Hérodotosz az ie. V. századi görög-perzsa háborúról szóló írásában is megemlíti. Demaratusz a perzsa Szúzában élt, így hamar értesült Xerxész Spárta elleni terveiről. Ezért – noha számkivetett volt – egy fatáblára véste figyelmeztető üzenetét, melyet aztán viasszal vont be. Ezt hívják szteganográfiának, amely a görög ***steganos*** (takar) és ***graphein*** (írni) szavak összetétele. Az elektronikus világban a szöveget gyakran képek, hangok bitjei közé rejtik, ám meg kell említeni, hogy természetesen megfelelő módszerek léteznek a felismerésükre is. A háború nyilván szélsőséges példa, ám a gyakorlat azt mutatja, hogy levelét azért mindenki borítékba teszi – noha gyenge védelem, mégis megóv a kíváncsi szemektől.

Ezzel párhuzamosan fejlődött a kriptográfia, melynek nevét a görög ***kryptos*** (rejtett) szóra vezethetjük vissza. Itt már nem magának az írásnak az elrejtése volt a cél, hanem az írás értelmének, üzenetének a kódolása. Az írást valamilyen visszafordítható eljárással érthetetlenné tették, majd ezt küldték el.

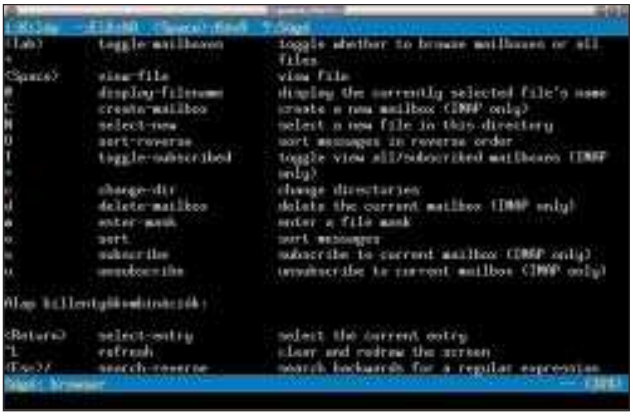
Az eljárás magában hordozta a kulcsot is, és az volt az előnye, hogyha az elküldött szöveget „elcsípték”, akkor is érthetetlen maradt. A kriptográfia alapvetően két módszert használ: a helyettesítést és az átrendezést, pontosabban egy adott kulcs segítségével egyértelműen helyettesíti az üzenet betűit, illetve egy előre rögzített szabály alapján cseréli fel azokat egymás között. Ismét következzék egy érdekes történelmi példa: az ie. VI. században a Kámaszutra szerint a nőknek 45. megszerzendő tudásként a ***mlecchita-vikalpa***-t, azaz a titkosírást kellett elsajátítaniuk. Egy javasolt módszer szerint az új ábécét az ábécé betűinek véletlenszerű társítása adta meg.

A fentiek alapján már mindenki tudna titkosítani, ám maradt még egy gond: a kulcs eljuttatása a fogadó félhez, hogy el is tudja olvasni az üzenetet. Itt lép be a nyilvános kulcsú kriptográfia, melynek lényege, hogy a kulcsot, pontosabban annak egy részét megosztjuk másokkal. A titkosítókulcsot két csoportra bontjuk: titkosra és nyilvánosra. Hogy ennek mi az értelme, egy példán keresztül szemléltetem.

Tegyük fel, hogy titkosított üzenetemet úgy küldöm el, hogy beteszem egy ládába, amit lelakatolok. A kulcsot megtartom magamnak. A címzett, amikor megkapta, ráteszi a saját lakatját is, és visszaküldi nekem. Ekkor én leszedem a saját lakatomat és visszaküldöm a ládát. Végül a címzett mindössze leszedi a saját lakatját és hozzáfér a levélhez. A „kulcsot” ezekután úgy érdemes elképzelni, hogy egy lakatból és annak kulcsából áll: a lakat legyen nyilvános, a kulcs pedig titkos. Azért kedvező, ha a lakat a nyilvános, mert előre szét lehet szórni a címzettek között. A feladó csak felteszi a címzett lakatját a ládára – a lakat ugyanis mindenki által könnyen zárható –, de (elvileg) csak a saját kulcsával nyílik, amit természetesen egyedül a címzett birtokol. Ha ilyen virtuális lakatunk lenne, könnyű lenne lemásolni és szétosztani a leendő feladók között. Nos, léteznek ilyen matematikai virtuális lakatok, amelyek az egyik irányban könnyen zárnak, ám nehéz kitalálni, milyen volt a hozzá tartozó kulcs. Ezek a módszerek leggyakrabban a véletlen (illetve a véletlenszerű) számokra és a faktorizációra épülnek.

Fontos még beszélnünk a hitelességről és az aláírásról. Amennyiben a fenti módszer segít abban, hogy a két fél megőrizze a titkát, ugyanúgy felhasználható a hitelesség garantálására is. Ehhez szavatolni kell, hogy a lakatok tulajdonosai valóban azok legyenek, akikre mi gondolunk. Ekkor az úgynevezett

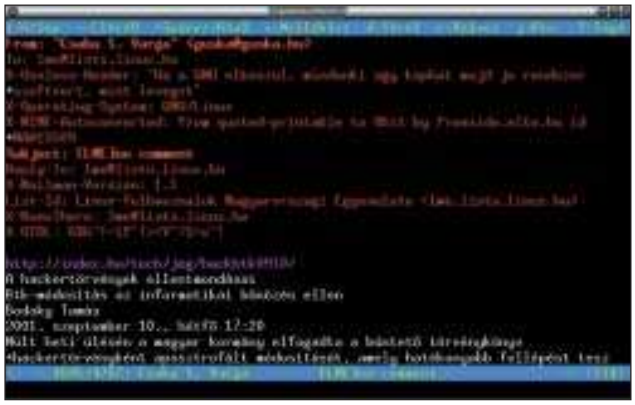
Mutt-rendezés, -rendszerezés



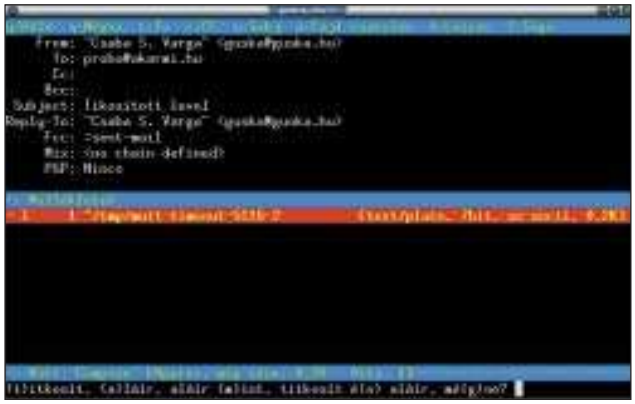
Mutt súgó

bizalmi háló használatos: a felek aláírják egymás nyilvános kulcsait, ha meggyőződtek a valódiságáról. Ezekután gondoljuk meg: ha valaki aláírta az én kulcsomat (mert ismer), akkor egy esetleges feladó, aki megbízik az ismerősömben, bízhat benne, hogy az én nyilvános kulcsomat használja. Természetesen a fenti viszonyokat igen összetetté lehet tenni, innen ered a háló elnevezés.

A fentiek összessége bonyolult rendszert képez, melynek alkalmazása különleges odafigyelést igényel, ez pedig ellentétes a könnyű használhatóság igényével. Néha olyan körülményes, hogy nem éri meg vele bajlódni, ám többnyire ilyenkor keletkeznek a biztonsági rések, hiszen lusták vagyunk és kódolást sem használunk. Ebben segít a Mutt levelezőprogram



Levél nézet



Mutt levelező és a GP menü

és a GnuPG programpáros: összekapcsolásukkal könnyen használható, olcsó levelezőrendszert hozhatunk létre, amely egy-két gombnyomásnyi közelségbe hozza a biztonságos levelezést. Nézzük, miképpen!

A Mutt levelezőprogramról mindenki úgy hiszi, hogy pilótavizsgás rendszer, de az első lépések megtétele után rá kell jönnünk, hogy bár a kezdet itt is nehéz, a továbbfejlődés könnyű és szinte határtalan. Elsőként határozzuk meg, hogy levelezőprogramunkban milyen szolgáltatásokra tartunk igényt:

- legyen könnyen kezelhető,
- öt különböző levélcímemet (család, munka stb.) kezelje,
- a beérkező leveleket válogassa – bizonyos szűrési feltételek szerint – külön könyvtárba,
- legyen belőle könnyen elérhető a GNUPGP,
- használja kedvenc szerkesztőmet és fájlnézegetőmet.

Általában majd minden Linux-terjesztésnek része a Mutt, a GPG, a Fetchmail és a Procmail csomagok, vagy forrásban hozzáférhetők az Interneten.

A Mutt első nekifutásra valóban nagyon keveset tudó alkalmazás, de ez csak azért van így, mert az alapbeállítás az úgynevezett „fapados” beállításokat tartalmazza. A telepítés után első lépésként saját könyvtárunkban a .muttrc-t kell létrehoznunk, mivel ez az állomány fog tartalmazni minden olyan apró beállítást, amelyet a program indulásakor mindig be fog olvasni, és ezek szerint fogjuk használni. Fenti igényeink megvalósításához a következő .muttrc szükséges:

```
#
# System-wide configuration file for Mutt
```

A színbeállítások

```
color hdrdefault red default
color quoted brightblue default
color signature red default
color indicator brightyellow red
color error brightred default
color status yellow blue
color tree magenta default
# the thread tree in the index menu
color tilde magenta default
color message brightcyan default
color markers brightcyan default
color attachment brightmagenta default
color search default green
# how to hilite search patterns in the pager

color header brightred default
➥^(From|Subject):
color body magenta default
➥"(ftp|http)://[^ ]+" # point out URLs
color body magenta default [-a-z_0-9.]+
➥@[-a-z_0-9.]+     # e-mail addresses
color underline brightgreen default
```

```
#
set folder=~/mail
```

Értelemszerűen itt határozzuk meg, hol legyen a kiindulási könyvtár.

```
set editor="vi"
```

Ez a kedvenc szerkesztőnk, amit a levelek megírásához szeretnénk használni.

```
set charset="iso-8859-2"
```

A levelek kódolása írásnál-olvasásánál egyaránt használatos.

```
set hostname="levelezo.munkallomas.hu"
```

A gépünk neve.

```
set sort=date-received
```

A beérkezés dátuma szerint rendezzük a leveleket, ami menet közben dinamikusan változtatható.

```
set spoolfile='~/mail/Inbox'
```

A bejövő levelek helye, illetve annak megadása, hogy az alkalmazás induláskor mely levélcímünkhöz tartozó leveleket töltse be.

```
set tmpdir=/home/guska/tmp
```

Ez adja meg, hogy a szerkesztőnk hova rakja az úgynevezett piszkozatot. Ennek akkor van nagy szerepe, ha egy esetleg bekövetkező áramszünet miatt az éppen megkezdett levelet szeretnénk megkeresni, továbbá azért is célszerű beállítani ezt a szolgáltatást, mert ellenkező esetben az alapértelmezés szerinti */tmp*-t használja, amelybe esetleg más felhasználók is beleláthatnak, és leveleink egy részét el is olvashatják.

```
my_hdr X-Operating-System: GNU/Linux
```

Megadhatjuk, hogy a levelező a levél fejlécében található operációsrendszer-meghatározásba mit is írjon. Ide érdemes nem túl sok adatot eláruló megjegyzést elhelyezni, ugyanis sokan rosszul értelmezett büszkeségből szinte a teljes rendszer jellemzését megadják, például: GNU/Linux Debian Woody, 2.4.9 rendszermag, Mutt 1.2 stb. Ezek az adatok gyengítik védelmi vonalainkat, hiszen ha például kijön egy ismert biztonsági hiba a 2.4.9-es rendszermagra, mi máris elárultuk, hogy azt használjuk.

© Kiskapu Kft. Minden jog fenntartva

© Kiskapu Kft. Minden jog fenntartva

Színbeállítások

Listánkon jól látható, hogy szinte minden mezőhöz színeket rendelhetünk, de csak a lényeges elemeket érdemes külön színnel ellátni, mert a túl színes levelező egy idő után átláthatatlan.

```
my_hdr From: Csaba S. Varga <guska@guska.hu>
my_hdr Reply-To: Csaba S. Varga <guska@guska.hu>
set signature="~/.signature"
folder-hook . my_hdr From: Csaba S. Varga
➥<guska@guska.hu>
folder-hook . my_hdr Reply-To: Csaba S. Varga
➥<guska@guska.hu>
folder-hook . my_hdr Cc: <>
folder-hook . my_hdr Bcc: <>
folder-hook . set signature="~/.signature"
```

Ezek a sorok mondják meg a levelezőprogramnak, hogyha egyszerűen csak a Postafiókból (Inbox), vagy valamilyen más, külön meg nem határozott postafiókból kezdeményezünk levélírást, ki legyen a feladó, és az aláírást melyik fájlból vegye stb.

```
folder-hook =lme my_hdr From: Csaba S. Varga
➥<guska@gnu.hu>
folder-hook =lme my_hdr Reply-To: Csaba S. Varga
➥<guska@gnu.hu>
folder-hook =lme set signature="~/.signature-lme"
```

Ha a levélírást az LME levelezőfiókból kezdeményezem, a levél a gnu.hu-s címmel és más aláírófájl használatával megy ki. Így lehet például a céges levélcímet és a magánlevélcímet ugyanabban a levelezőben használni.

```
fcc-hook . +fcc-hook .
```

Minden Muttból küldött levelet tárolás céljából a „sent-mail” levelesládába fűzzön le. Ez a szolgáltatás nagyon hasznos lehet vitás levelek utólagos előkeresésekor.

```
set pgp_verify_sig=yes
set pgp_replyencrypt
set pgp_replysign
set pgp_timeout=600
```

Ezek a sorok tartalmazzák a PGP- és GPG-programokra vonatkozó meghatározást, amelyben jelszavunk időbeni érvényességét rögzítjük, továbbá ha valaki eleve titkosított levelet ír nekünk, akkor véletlenül se fordulhasson elő, hogy általános levélként továbbítjuk, illetve segítségével a válasz is önműködően kódolt lesz.

```
send-hook kisvakond 'set pgp_autosign=yes; set
pgp_autoencrypt=yes'
```

Amennyiben valakivel olyan levelezési kapcsolatban állunk, hogy minden egyes levelünk titkosított küldését követeli meg, érdemes használni, hiszen így a Kisvakond becenevű illetőnek úgy írhatunk levelet, hogy a rendszer elküldés előtt rákérdez GPG-jelszavunkra.

Az új 1.1-es Mutt-rendszerek GPG-kezeléséhez további sorok szükségesek, mivel azonban ezek igen hosszú bejegyzések és igazából csak a GPG számára értelmezhetők, érdemes őket a GPG-leírásból bemásolni (körülbelül 10–15 sor).

A Mutt innen olvassa be a címjegyzékünket, amelyet az adott formátumban kell elhelyezni. Ha a levelezőprogramban egy levélen állva megnyomjuk az A billentyűt, a címzett nevét és címét a .muttrc-hez fűzi.

A Debian Woodyban található 1.3.20-1-es Mutt már a magyar nyelvet is támogatja.

Procmail

A Mutt összes szolgáltatásának kényelmes eléréséhez érdemes a Procmail segédprogramot használni. Ez olyan alkalmazás, amely a levelezőkiszolgálótól átveszi a leveleket, amelyeket a megadott feltételek alapján szűrni is tud, majd az általunk megnevezett levelesládába rakja. A Muttnál itt is szükséges egy beállítóállomány, amelyet .procmailrc néven a könyvtárunkba kell helyezni. A .procmailrc csak akkor használható, ha a gépünkre mind levelezőkiszolgáló program, mind a Procmail segédprogram telepítve van.

A .procmailrc

```
SHELL=/bin/bash
```

Ez adja meg, milyen héjprogramot használunk.

```
MAILDIR=~/mail
```

A levelek könyvtárának a helye.

```
LOGFILE=$HOME/procmailrc.log
```

A könnyebb kezelhetőség érdekében érdemes naplózni, hiszen ez alapján megtudhatjuk, hogy egy-egy letöltéskor mely levelesládánkba hány levél érkezett.

```
:0
* ^(From|Cc|To|Reply\-To):.*haragosom@domainje.hu
/dev/null
```

Egy Procmail-bejegyzés általában három sorból áll, a :0 határozza meg, hogy egy szűrési feltétel fog következni, a * után pedig felsoroljuk, hogy a levél milyen mezőjében mit szeretnénk szűrni. Jelen esetben, ha a haragosom@domainje.hu-tól kapunk levelet vagy neki küldték, önműködően a nagy „Unix-iktatóba” kerül.

```
:0
* ^Reply-To: lme@lists.linux.hu
$MAILDIR/lme
```

A formai követelmények ugyanazok, csak jelen esetben a levelet nem eldobjuk, hanem levelesládába rakjuk.

```
:0 wc
* .*
$MAILDIR/mentes
```

Ha ez az utolsó sor a .procmailrc-ben, akkor minden bejövő levélből az irattárba is kerül, és az előtte meghatározott szűrésekből értelemszerűen nem kerül mentésre, csak ami túlhaladt azokon a szűrési feltételeken. Ez hasznos lehet, mert így a különböző levelezési listákat külön tárolhatjuk, valamint ***mentés*** nevű levelesládánkba csak a fontos levelek kerülnek. Mivel másolatról van szó, a levél a Postafiókba is eljut.

```
:0
* .*
$MAILDIR/Inbox
```

Ezzel a Postafiókot adjuk meg.

További hasznos segédprogram még a Procmailhez tartozó Mailstat , amely a procmail.log-ot a 2. listán (20. CD Magazin/Mutt) látható módon képes kiértékelni.

Természetesen mind a Procmail, mind a Mutt beállításai a fentieken kívül még számos kedvező lehetőséget nyújtanak. A bemutatott beállítások segítségével jól használható levelezést alakíthatunk ki, amelyet szinte a végtelenségig tovább lehet fejleszteni.

E cikkre a Free Document Licence vonatkozik.
➲ http://www.gnu.hu/fdl.html

Varga S. Csaba
(guska@guska.hu) A 1.1-es Slackware óta linuxozik. Kedvtelései közé tartozik a fotózás és Linux telepítése a PDA-kra. Legszívesebben a Gerecsében túrázik. Barátjával, *Somogyi (Jerry) Péterrel* együtt írta e cikket.