# BEZPEČNÁ E-MAILOVÁ KOMUNIKACE

## digitální podpis a šifrování pro každého


Zdeněk Maršík

-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.7 (MingW32)
Comment: Using GnuPG with Mozilla - http://enigmail.mozdev.org

iD8DBQFFgR3PrwKZw+Q6QN8RAh+vAKCrEfSe3o1fJA5/7LEYQIlerdrU2wCdE2kK
K3ZgTVZTcspPdVyRYAoSbJU=ZOQj
-----END PGP SIGNATURE-----

# Bezpečná e-mailová komunikace pomocí GnuPG ve WINDOWS i v LINUXu

*autor: Zdeněk Maršík © 2007*

## Obsah

## Úvod k bezpečné komunikaci

Proč používat bezpečnou komunikaci? Odpovím otázkou a proč ne?
Posílání emailů je podobné jako posílání pohlednice, tedy zpráva není nijak chráněna před zvědavci. Takovou zprávu lze přečíst, aniž by se o tom adresát nebo odesílatel dozvěděl. Ovšem zpráva chráněná programem GnuPG (GPG) je podobná dopisu v obálce, ale je samozřejmě lépe chráněna a je nemožné, aby takovou zprávu někdo nepovolaný přečetl. Pokud je ještě podepsána, tak máte jistotu, že pochází od odesílatele. Jistě si dovedete představit situaci, kdy zákazník si objedná nějaké zboží (software) u obchodníka a zaplatí na číslo účtu, které mu obchodník poslal e-mailem. Ovšem HACKER změnil číslo účtu na svůj účet a vesele inkasuje peníze. Na obrázku na straně 19 je **zašifrovaný a podepsaný email** tak, jak ho vidí oprávněný příjemce. Vše probíhá zcela automaticky bez větší námahy. Tento program je zdarma, bez certifikačních poplatků. Bezpečnost je na nejvyšší úrovni, tedy je to nerozluštitelné i pro CIA, FBI, MI5 ap. GPG je založen na **asymetrické kryptografii.** Tato je založena na **prvočíslech** a obtížnosti rozkladu součinu dvou prvočísel.
Používá soukromý a veřejný klíč dle standardu IETF RFC 2440 šifrování příloh dle RFC 3156. Veřejný klíč je možné zveřejnit na http://zmsoft.cz/gpg.html. Zde popisuji multiplatformní nástroje, lze je tedy používat ve WINDOWS i v LINUXu **zadarmo a legálně** i ke komerčním účelům (tedy zde není žádné omezení). Pomocí programu GnuPG lze také šifrovat soubory a vytvářet si tak zabezpečené zálohy apod. S GPG lze posílat **elektronické digitálně podepsané faktury**. E-mailem si nechat zasílat zůstatky na bankovních účtech, lékařské zprávy, výpisy z trestních rejstříků, výši vkladů pojištění atd.

Co budete potřebovat:

1. **GnuPG** - GPG (GNU Privacy Guard) vychází z PGP (Pretty Good Privacy). Jedná se o open-source programový balík pro bezpečnou komunikaci a předávání dat. Umožňuje šifrování a dešifrování, podepisování a kontrolu podpisu různých dat, třeba emailů. Internet: http://www.gnupg.org/
2. **THUNDERBIRD** - vynikající poštovní program (náhrada za outlook express), umožňuje psát v různých znakových sadách (např. UTF-8), pak se to v zahraničí správně zobrazuje (Internet http://www.czilla.cz/).
3. **ENIGMAIL** - doplněk k thunderbirdu pro šifrování pošty a správu klíčů.

GPG pracuje s kroužky na klíče! Má kroužek veřejných klíčů a kroužek s tajným klíčem.

# Co je digitální podpis

Digitální podpis je bezpečnostní mechanismus, který je nezfalšovatelný a je založen na asymetrické kryptografii. Digitální podpis zaručuje, že **zpráva nebyla změněna** a **autorem zprávy je opravdu osoba, která je podepsaná.** Byl vytvořen jako náhrada klasického ručního podpisu, který je pro každou osobu jedinečný, ale je zfalšovatelný. V prostředí Internetu při elektronické komunikaci je obtížné ověřit, zda odesílatelem zprávy je skutečně ten, který je napsán. Také není možno zjistit, zda informace v elektronické komunikaci nebyla pozměněna nebo dokonce někým odposlechnuta (přečtena). Digitální podpis (zaručený elektronický podpis) a šifrování řeší tuto problematiku a je moderní součástí profesionální elektronické komunikace.
Existují dva standardy:

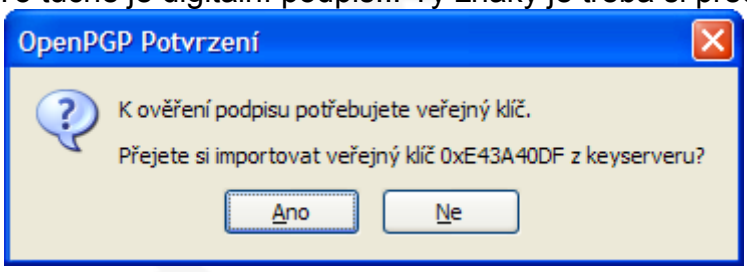
1. X.509 certifikáty
2. RFC 2440 certifikáty

GPG používá standard RFC 2440 a 3156 pro šifrování a digitální podpisy. Odesílatel podepíše zprávu svým klíčem (musí znát heslo). Příjemce zprávy ověří digitální podpis veřejným klíčem

odesílatele , který je veřejně dostupný na Internetu.
**Takže digitální podpis je „výtah zprávy“ SHA-1 až SHA-512 za použití tajného klíče odesílatele (podepsání) a veřejného klíče odesílatele (ověření podpisu) při použití asymetrické kryptografie. Příjemce musí pro ověření dig. podpisu získat veřejný klíč odesílatele!**

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

Dobrý den,
dne 12. 12. 2006 nám bylo zaplaceno za software. Pro plnou funkčnost je třeba se
zaregistrovat dle instrukcí na: http://zmsoft.cz/. Obvykle to provádí správce sítě,
pokud jím nejste vy, tak prosím, přepošlete tento email také správci počítačové sítě.

Výpis z účtu 1115011026/0800
Číslo účtu příjemce 1115011026/0800
Přeji hezký den

Zdeněk Maršík
Květná 1843
688 01 Uherský Brod

Tento email je digitálně podepsán veřejným klíčem:
ID (keyID): E43A40DF
UID (User ID): Zdenek Marsik (Urbo: Uhersky Brod)
Otisk prstu (fingerprint): A858 94F9 5D73 DD88 D8D0 D791 AF02 99C3 E43A 40DF

-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.5 (MingW32)
Comment: Using GnuPG with Mozilla - http://enigmail.mozdev.org

iD8DBQFFgR3PrwKZw+Q6QN8RAh+vAKCrEfSe3o1fJA5/7LEYQlIerdrU2wCdE2kK
K3ZgTVZTcspPdVyRYAoSbJU=ZOQj
-----END PGP SIGNATURE-----
```

To tučné je digitální podpis!!! Ty znaky je třeba si představit jako čísla, každý znak jedno číslo.



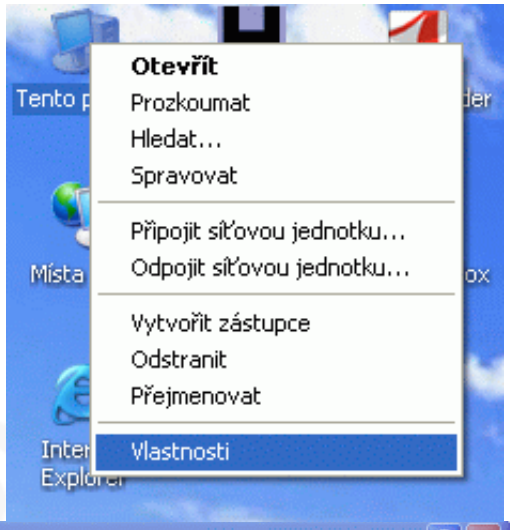

Z výše uvedeného vyplývá, že **nejbezpečnější** formou elektronické komunikace je **zašifrování zprávy a její digitální podepsání.** Aby mohla být zpráva u odesílatele zašifrována, je potřeba mít veřejný klíč příjemce. K podepsání slouží odesílatelův tajný klíč. Aby mohla být digitálně podepsaná zpráva ověřena (ověření správnosti digitálního podpisu) u příjemce, je potřeba, aby příjemce měl k dispozici veřejný klíč odesílatele. Proto je nejlepší svůj veřejný klíč umístit na Internetu na adresách k tomu určených, například http://zmsoft.cz/gpg.html. Tuto adresu je též možno využít pro vyhledávání veřejných klíčů nejen z ČR ale z celého světa. Pak při potřebě veřejného klíče se poštovní klient automaticky zeptá a stáhne potřebný veřejný klíč, viz obrázek.

---

ZM - SOFT
Zdeněk Maršík
tvorba software
prodej výpočetní techniky
Květná 1843, 688 01 Uherský Brod
Tel. 0603 506 407

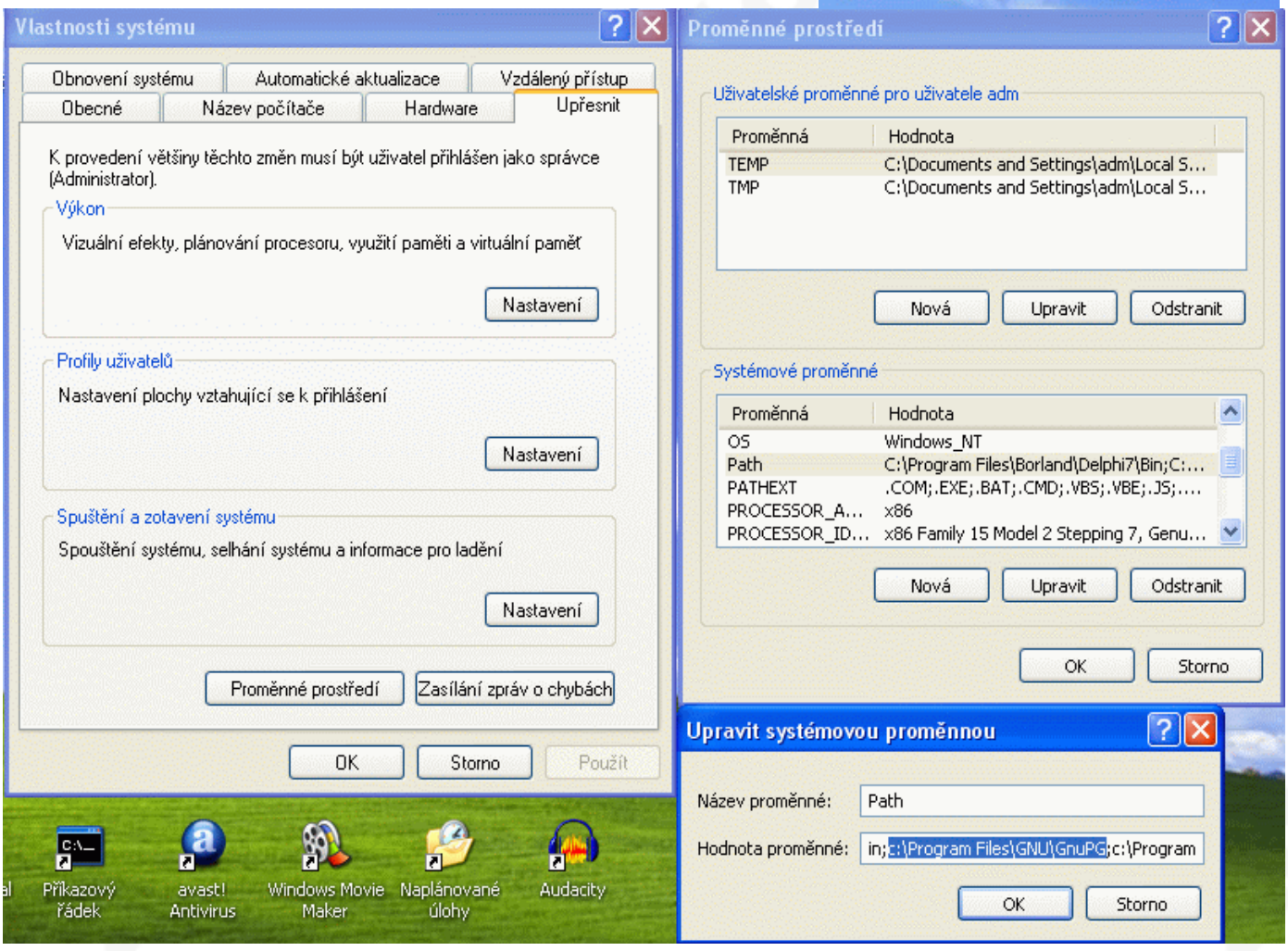
**Toto není digitální podpis.** Není to ani elektronický podpis, ani zaručený elektronický podpis. Je to pouze naskenované razítko a podpis, které jsou vloženy do dokumentu. Pokud jste se domnívali, že tohle digitální podpis, tak je tato kniha pro vás již nyní obrovským přínosem.

## GnuPG - gpg (Internet http://www.gnupg.org/)




1. Zkuste, zda GPG=OPENGP již je nainstalované příkazem z konzoly: `gpg --help`
2. Pokud není GNUPG nainstalováno, stáhněte a nainstalujte si jej z www.gnupg.org pro svůj operační systém
3. po instalaci napište v konzoli `gpg --help`, zobrazí se nápověda (ve WINDOWS je potřeba nastavit cestu k gpg.exe)

Nastavení cesty ve WINDOWS-XP se provede pod účtem administrator (nebo účet s právy administrators) takto:
-pravým tlačítkem myši na TENTO POČÍTAČ, vlastnosti, upřesnit, proměnné prostředí, path, upravit.



## Tvorba páru klíčů

V příkazovém řádku (ve WIN-XP START, SPUSTIT, napsat cmd) zadejte příkaz:

```
gpg --gen-key

gpg (GnuPG) 1.4.7; Copyright (C) 2006 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

Prosím, vyberte druh klíče, který chcete:
   (1) DSA a ElGamal (implicitní)
   (2) DSA (pouze pro podpis)
   (5) RSA (pouze pro podpis)
Váš výběr? 1
Pár DSA klíčů DSA dlouhý 1024 bitů.
klíč ELG-E může mít délku v intervalu 1024 až 4096 bitů.
Jakou délku klíče si přejete? (2048) 4096
Požadovaná délka klíče je 4096 bitů.
Prosím určete, jak dlouho by klíč měl platit.
         0 = doba platnosti klíče není omezena
      <n>  = doba platnosti klíče skončí za n dní
      <n>w = doba platnosti klíče skončí za n týdnů
      <n>m = doba platnosti klíče skončí za n měsíců
      <n>y = doba platnosti klíče skončí za n let
Klíč je platný pro? (0) 730
Platnost klíče skončí v Út  3. březen 2009, 18:53:36 CET
Je to správně (a/N)? a

Aby bylo možné rozpoznat Váš klíč, musíte znát identifikátor uživatele;
program jej složí z Vašeho jména a příjmení, komentáře a e-mailu
v tomto tvaru:
    "Magda Prochazkova (student) <magda@domena.cz>"

Jméno a příjmení: Magda Procházková
E-mailová adresa: prochi@atlas.cz
Komentář: Ječná 1967, UHERSKÝ BROD, ČOP:EK 564789
Používáte znakovou sadu `ISO-8859-2'.
Zvolil(a) jste tento identifikátor uživatele:
    "Magda Procházková (Ječná 1967, UHERSKÝ BROD, ČOP:EK 564789) <prochi@atlas.cz>"

Změnit (J)méno, (K)omentář, (E)-mail, (P)okračovat dál nebo (U)končit program? P
Pro ochranu Vašeho tajného klíče musíte zadat heslo.

gpg: gpg-agent není v tomto sezení dostupný
Musíme vytvořit mnoho náhodných bajtů. Během vytváření můžete
provádět nějakou jinou práci na počítači (psát na klávesnici, pohybovat myší,
používat disky); díky tomu má generátor lepší šanci získat dostatek entropie.
+++++++++++++..++++++++++++++++++++++++.+++++.++++++++++++++++++++++++++++++.++++++
++++++++++++++++++.++++++++++++++++++++++++++++>++++++++++.....+++++
Musíme vytvořit mnoho náhodných bajtů. Během vytváření můžete
provádět nějakou jinou práci na počítači (psát na klávesnici, pohybovat myší,
používat disky); díky tomu má generátor lepší šanci získat dostatek entropie.
+++++.....+++++++++++++++..++++++++++.+++++.++++++++++...+++++++++++++++.+++++.++++++
++++.+++++.+++++++++++++++...+++++.+++++.+++++.++++++++++.++++++++++++++++++++++++++++
++.....+++++.++++++++++>++++++++++++++++++++++++++++++++++++++..+++++....+++++>+++++
>+++++>..+++++>+++++>.+++++.....................................................
.....................................................
.....................................................
.<+++++....>+++++<+++++....>+++++......<.+++++.....................>+++++<+++++.......
>...+++++.....................................................>.+++++<+++++.............>+++
++<.+++++.....................................................
.....................>+++++
```

```
<+++++.....................................<.+++++....................>+++++<.+++++..
.....................................>+++++..............................<.+++++..
....>+++++<+++++................>+++++.......<..+++++.....>+++++.....>+++++.........
.......................<.+++++.....................................................
.....................................................................................
.....................................................<...+++++..>...+++++.............
.....................................................................>+++++.....
.....................................<+++++...........>+++++.........<....+++++.
.....................................................................................
.........................................<.+++++.....>..+++++.............+++
++^^^
gpg: klíč 68A4C546 označen jako absolutně důvěryhodný.
veřejný a tajný klíč byly vytvořeny a podepsány.

gpg: kontroluji databázi důvěry
gpg: požadováno 3 částečné důvěry a 1 úplné důvěry, model PGP
gpg: hloubka: 0  platných:   1  podepsaných:   0  důvěra: 0-, 0q, 0n, 0m, 0f, 1u
gpg: další kontrola databáze důvěry v 2009-03-03
pub   1024D/68A4C546 2007-03-04 [platnost skončí: 2009-03-03]
      Fingerprint klíče = 0A8F 9B44 9A7C 7F10 47BF  9093 A37A 4421 68A4 C546
uid   Magda Procházková (Ječná 1967, UHERSKÝ BROD, ČOP:EK 564789) <prochi@atlas.cz>
sub   4096g/8179B34A 2007-03-04 [platnost skončí: 2009-03-03]
```

## Vložení fotky 91 x 114 pixelů (.jpg)

`gpg --edit-key 68A4C546 <Enter>`
`addphoto <Enter>` - přidání fotky
`save <Enter>` - uložení změn

*Poznámka:*
*ČOP = číslo občanského průkazu*
*Ihned po vytvoření páru klíčů je třeba vytvořit revokační certifikát:*

## Základní pojmy

ID = číselné označení klíče – bývá označováno též jako **Key ID** (v hexadecimálním tvaru)
příklad: 1CE0C630
UID = **User ID** - identifikační popis uživatele, například jméno + (adresa) + <email>
příklad: Werner Koch (Potocni 1350, Praha 1) <dd9jn@gnu.org>
FINGERPRINT = otisk prstu - jednoznačný a nezfalšovatelný výtah zprávy SHA1 nebo až SHA2
příklad: 7B96 D396 E647 1601 754B  E4DB 53B6 20D0 1CE0 C630
g = šifra ELGamal
D = šifra DSA
R = šifra RSA

## Vytvoření revokačního (zneplatňujícího) certifikátu

Nyní je doporučeno si vytvořit revokační certifikát, který si uložíme na bezpečné místo mimo PC nebo notebook. Slouží k zneplatnění (odvolání) veřejného klíče. K zneplatnění dojde také při vypršení doby platnosti. Při pádu systému se ztrátou všech dat pak můžeme odvolat (zneplatnit veřejný klíč uložený na internetu). Postup vytvoření zneplatňujícího certifikátu:

a) zjištění ID vlastního veřejného klíče: `gpg -K`,

sec 1024D/**D1A4DF62** 2006-01-07 [platnost skončí: 2008-01-07]
uid Zdenek Marsik (Granda Urso) <mzdenek@post.cz>
ssb 4096g/CB4B8E9D 2006-01-07

Výborně, máme ID veřejného klíče D1A4DF62, které patří našemu páru klíčů, zadáme příkaz pro vygenerování revokačního certifikátu:

```
gpg --output D1A4DF62_revoke.asc --gen-revoke D1A4DF62

Vytvořit pro tento klíč revokační certifikát? (a/N) a
Prosím vyberte důvod revokace:
  0 = Důvod nebyl specifikován
  1 = Klíč byl zkompromitován
  2 = Klíč je nahrazen
  3 = Klíč se již nepoužívá
  Q = Zrušit
(Pravděpodobně zda chcete vybrat 1)
Vaše rozhodnutí? 1
Můžete vložit další popis. Ukončete prázdným řádkem:
>
Důvod revokace: Klíč byl zkompromitován
Je důvod revokace vybrán správně? (a/N) a

Musíte znát heslo, abyste odemknul(a) tajný klíč pro
uživatele: "Zdenek Marsik (Granda Urso) <mzdenek@post.cz>"
délka 1024 bitů, typ DSA, klíč D1A4DF62, vytvořený 2006-01-07

nařízen výstup do formátu ASCII.
Revokační certifikát byl vytvořen.
```

**Prosím přeneste jej na médium, které můžete dobře schovat.** Pokud se k tomuto certifikátu dostane nepovolaná osoba, může zneplatnit Váš klíč. Je rozumné tento certifikát vytisknout a schovat jej pro případ, že médium s certifikátem přestane být čitelné. Ale pozor: Tiskový subsystém na Vašem počítači může ukládat data určená k tisku a zpřístupnit je jiným uživatelům!

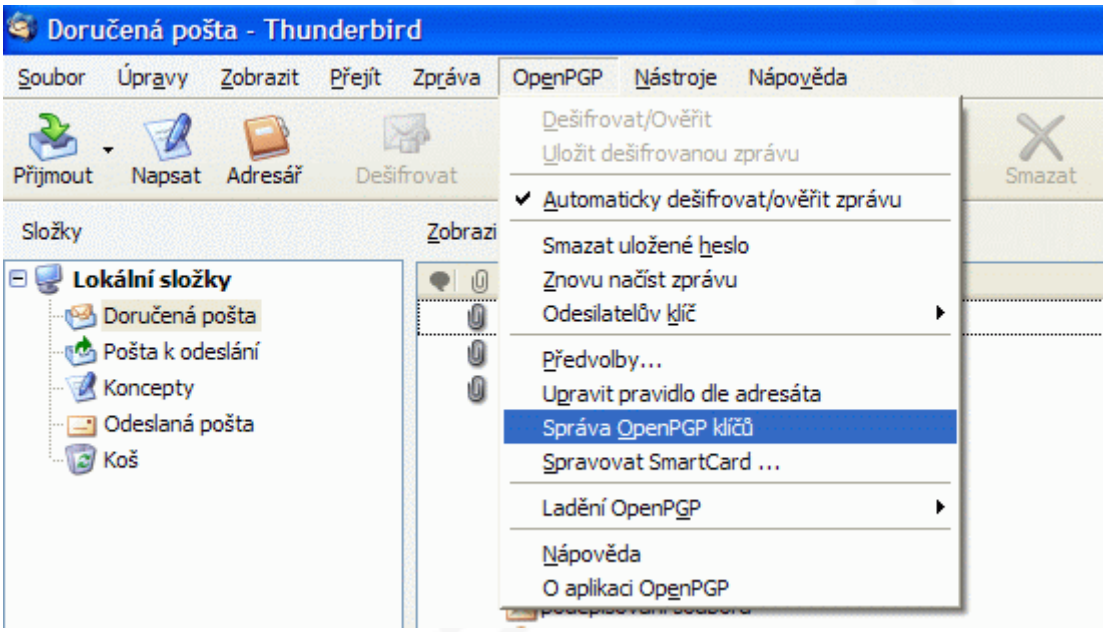
Po ztrátě soukromého klíče nebo při potřebě změnit svůj veřejný klíč se dále postupuje takto:

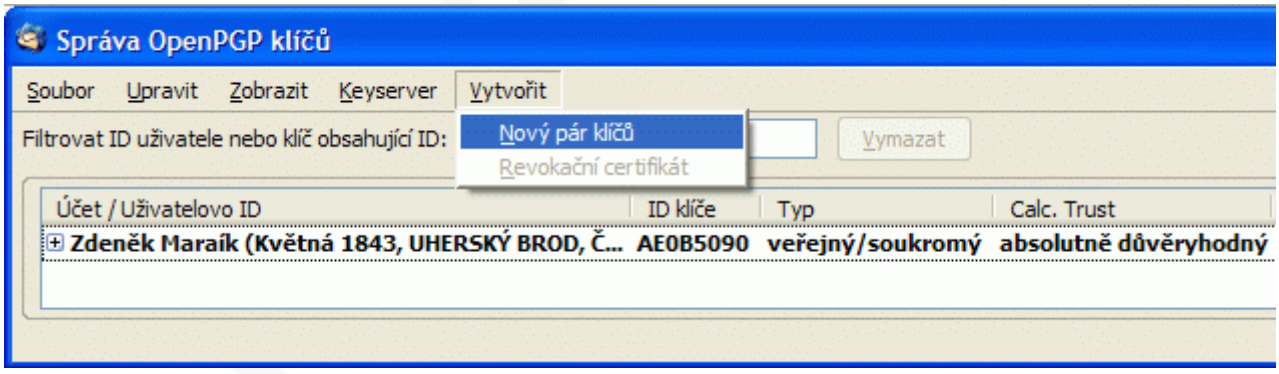
`gpg --import revoke_zdenek_marsik.asc`
`gpg --output zdenek_marsik.asc --export -a D1A4DF62`
nahrát tento revokační veř. klíč na stránkách http://zmsoft.cz/gpg.html
smazat tajný klíč příkazem `gpg --delete-secret-keys D1A4DF62`
smazat veř. klíč příkazem `gpg --delete-keys D1A4DF62`
vytvořit nový pár klíčů příkazem `gpg --gen-key`

Pro vytvoření zneplatňujícího certifikátu je potřeba **tajný klíč** a **znát heslo** k tomuto klíči. Pokud jedna z těchto podmínek nebude splněna tak se nepodaří zneplatnit (revokovat) váš veřejný klíč. Proto je doporučováno vytvoření zneplatňujícího certifikátu hned po vytvoření páru klíčů. Po ukradení NOTEBOOKu nebo při zapomenutí hesla je pak možno váš veřejný klíč zneplatnit (odvolat, revokovat).

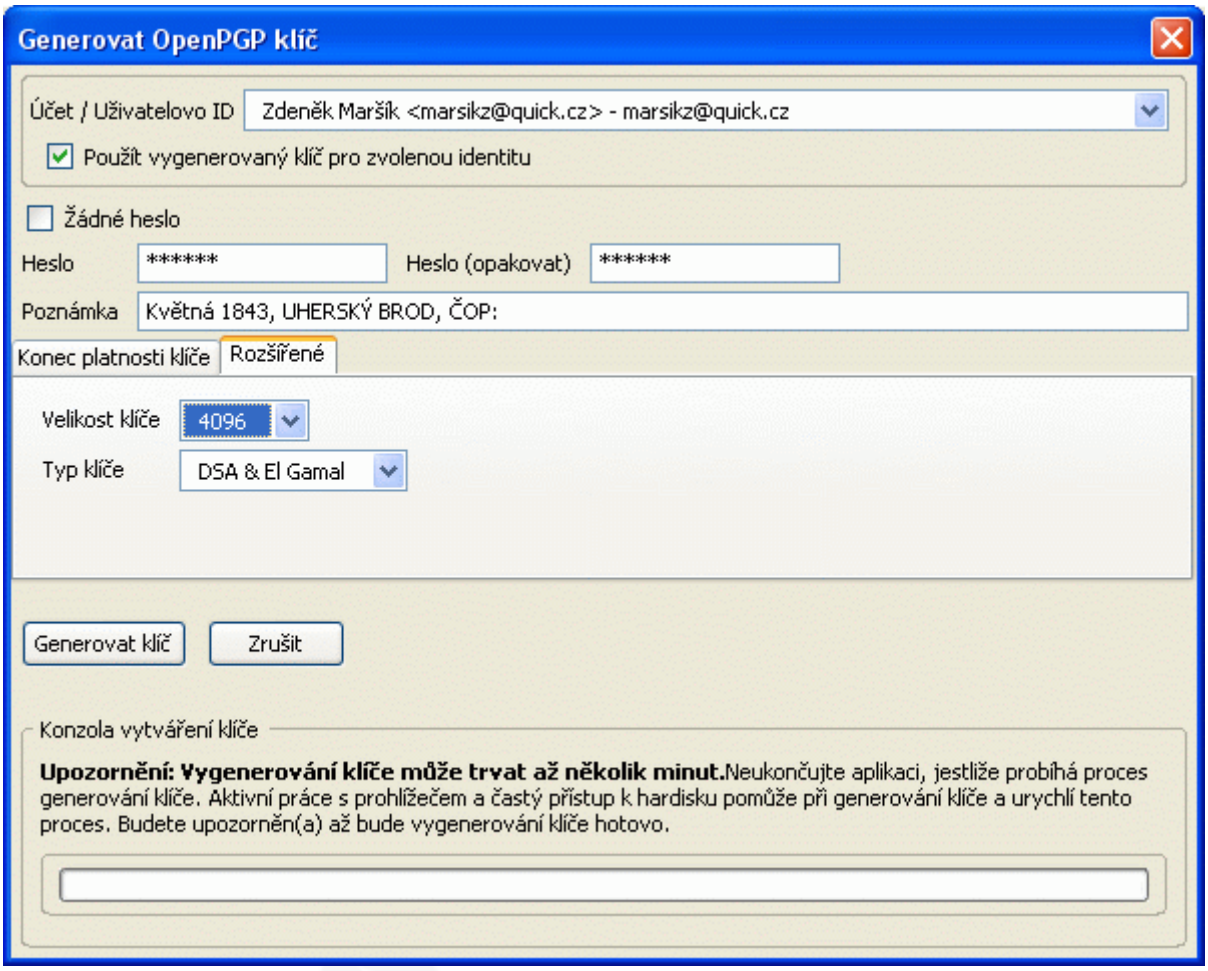
## Tvorba páru klíčů doplňkem ENIGMAIL

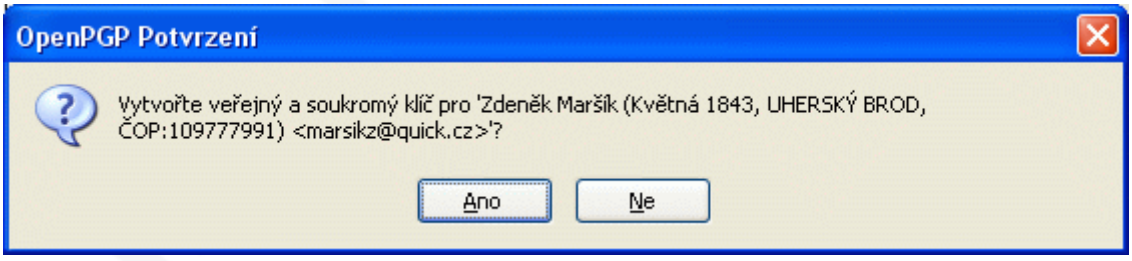
Taky lze generovat pár klíčů v grafickém rozhraní, pomocí doplňku thunderbirdu ENIGMAIL. Zvolíme v menu, OpenPGP, správa OpenPGP klíčů, menu, vytvořit:

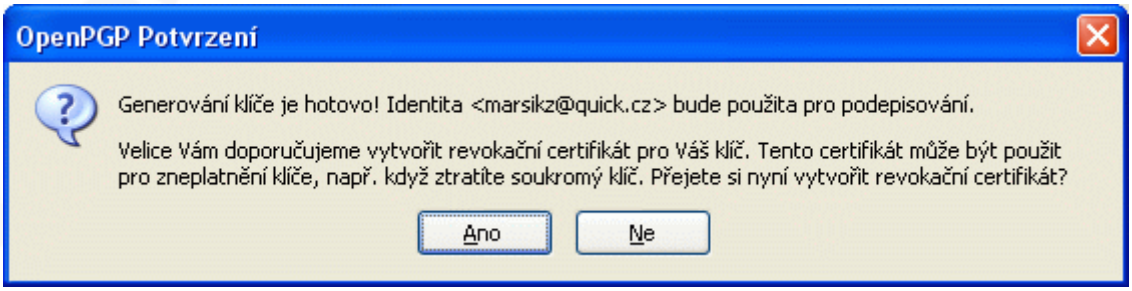


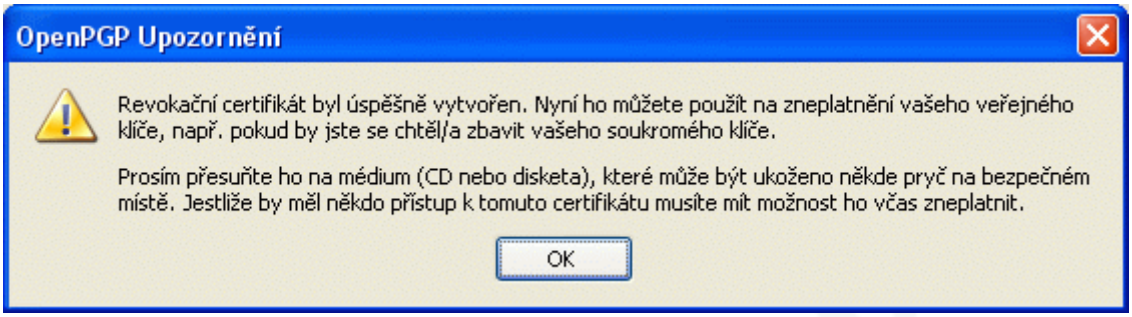



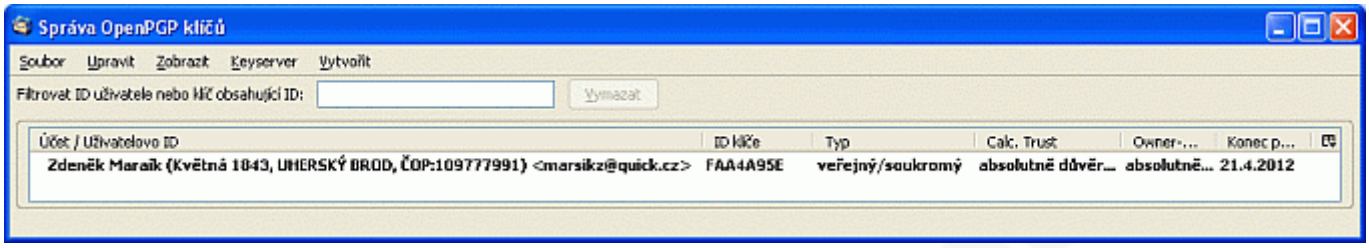

Zadáme stejné údaje, které se zadávají při generování páru klíčů přímo programem gpg v příkazové řádce. Myslím, že následující obrázky jsou dostatečně výmluvné a není třeba dalšího komentáře.


Generovat OpenPGP klíč
Účet / Uživatelovo ID
Zdeněk Maršík <marsikz@quick.cz> - marsikz@quick.cz
Použít vygenerovaný klíč pro zvolenou identitu
Žádné heslo
Heslo
******
Heslo (opakovat)
******
Poznámka
Květná 1843, UHERSKÝ BROD, ČOP:
Konec platnosti klíče
Rozšířené
Velikost klíče
4096
Typ klíče
DSA & El Gamal
Generovat klíč
Zrušit
Konzola vytváření klíče
Upozornění: Vygenerování klíče může trvat až několik minut.Neukončujte aplikaci, jestliže probíhá proces generování klíče. Aktivní práce s prohlížečem a častý přístup k hardisku pomůže při generování klíče a urychlí tento proces. Budete upozorněn(a) až bude vygenerování klíče hotovo.


OpenPGP Potvrzení
Vytvořte veřejný a soukromý klíč pro 'Zdeněk Maršík (Květná 1843, UHERSKÝ BROD, ČOP:109777991) <marsikz@quick.cz>'?
Ano
Ne


OpenPGP Potvrzení
Generování klíče je hotovo! Identita <marsikz@quick.cz> bude použita pro podepisování.
Velice Vám doporučujeme vytvořit revokační certifikát pro Váš klíč. Tento certifikát může být použit pro zneplatnění klíče, např. když ztratíte soukromý klíč. Přejete si nyní vytvořit revokační certifikát?
Ano
Ne


OpenPGP Upozornění
Revokační certifikát byl úspěšně vytvořen. Nyní ho můžete použít na zneplatnění vašeho veřejného klíče, např. pokud by jste se chtěl/a zbavit vašeho soukromého klíče.
Prosím přesuňte ho na médium (CD nebo disketa), které může být ukoženo někde pryč na bezpečném místě. Jestliže by měl někdo přístup k tomuto certifikátu musíte mít možnost ho včas zneplatnit.
OK




## Příkazy gpg:

```
 -s, --sign [soubor]           vytvořit podpis
     --clearsign [soubor]      vytvořit podpis v čitelném dokumentu
 -b, --detach-sign             vytvořit podpis oddělený od dokumentu
 -e, --encrypt                 šifrovat data
 -c, --symmetric               šifrování pouze se symetrickou šifrou
 -d, --decrypt                 dešifrovat data (implicitně)
     --verify                  verifikovat podpis
     --list-keys               vypsat seznam klíčů
     --list-sigs               vypsat seznam klíčů a podpisů
     --check-sigs              vypsat a zkontrolovat podpisy klíčů
     --fingerprint             vypsat seznam klíčů a fingerprintů
 -K, --list-secret-keys        vypsat seznam tajných klíčů
     --gen-key                 vytvořit nový pár klíčů
     --delete-keys             odstranit klíč ze souboru veřejných klíčů
     --delete-secret-keys      odstranit klíč ze souboru tajných klíčů
     --sign-key                podepsat klíč
     --lsign-key               podepsat klíč lokálně
     --edit-key                podepsat nebo modifikovat klíč
     --gen-revoke              vytvořit revokační certifikát
     --export                  exportovat klíče
     --send-keys               exportovat klíče na server klíčů
     --recv-keys               importovat klíče ze serveru klíčů
     --search-keys             vyhledat klíče na serveru klíčů
     --refresh-keys            aktualizovat všechny klíče ze serveru klíčů
     --import                  importovat/sloučit klíče
     --card-status             vytisknout stav karty
     --card-edit               změnit data na kartě
     --change-pin              změnit PIN karty
     --update-trustdb          aktualizovat databázi důvěry
     --print-md algo [soubory]  vypiš hash
     --encrypt-files -r KeyID * zašifruje všechny soubory v adresáři
     --dump-options            výpis všech příkazů a nastavení

Možnosti:

 -a, --armor                   vytvoř výstup zakódovaný pomocí ASCII
 -r, --recipient JMÉNO         šifrovat pro JMÉNO
 -u, --local-user              použít tento id uživatele pro podepsání
                                nebo dešifrování
 -z N                          nastavit úroveň komprimace N (0 - žádná
                                komprimace)
     --textmode                použít kanonický textový mód
 -o, --output                  použít jako výstupní soubor
 -v, --verbose                 s dodatečnými informacemi
 -n, --dry-run                 neprovádět žádné změny
 -i, --interactive             vyžádat potvrzení před přepsáním
     --openpgp                 použít chování striktně podle OpenPGP
     --pgp2                    generovat zprávu kompatibilní s PGP 2.x

(Použijte manuálové stránky pro kompletní seznam všech příkazů a možností)

 -se -r Bob [soubor]        podepsat a zašifrovat pro uživatele Bob
 --clearsign [soubor]       vytvořit podpis čitelného dokumentu
```

```
 --detach-sign [soubor]     vytvořit podpis oddělený od dokumentu
 --list-keys [jména]        vypsat klíče
 --fingerprint [jména]      vypsat fingerprinty
```

## Interaktivní režim:

**`gpg -K`** = zjištění ID vašeho veřejného klíče:
sec 1024D/D1A4DF62 2006-01-07 [platnost skončí: 2008-01-07]
uid Zdenek Marsik (Granda Urso) <mzdenek@post.cz>
ssb 4096g/CB4B8E9D 2006-01-07

**`gpg --edit-key d1a4df62`**

```
gpg (GnuPG) 1.4.7; Copyright (C) 2006 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

Tajný klíč je dostupný.
pub  1024D/D1A4DF62  vytvořen: 2006-01-07  platnost skončí: 2008-01-07  použití: CS
                     důvěra: absolutní     platnost: absolutní
sub  4096g/CB4B8E9D  vytvořen: 2006-01-07  platnost skončí: 2008-01-07  použití: E
[  absolutní ] (1). Zdenek Marsik (Granda Urso) <mzdenek@post.cz>

Příkaz> help
quit        ukončit toto menu
save        uložit a ukončit
help        ukázat tuto pomoc
fpr         vypsat fingerprint klíče
list        vypsat seznam klíčů a id uživatelů
uid         vyberte identifikátor uživatele N
key         vyberte podklíč N
check       kontrolovat podpisy
sign        podepsat vybrané ID uživatele [  níže jsou uvedeny relevantní příkazy]
lsign       podepsat vybrané uživatelské ID lokálně
tsign       podepsat vybrané uživatelské ID důvěryhodným podpisem
nrsign      podepsat vybrané uživatelské ID nerevokovatelným podpisem
adduid      přidat identifikátor uživatele
addphoto    přidat fotografický ID
deluid      smazat vybrané ID uživatele
addkey      přidat podklíče
addcardkey  přidat klíč na kartu
keytocard   přesunout klíč na kartu
bkuptocard  přesunout záložní klíč na kartu
delkey      smazat vybrané podklíče
addrevoker  přidat revokační klíč
delsig      smazat podpisy z vybraných uživatelských ID
expire      změnit datum expirace pro klíč nebo vybrané podklíče
primary     označit vybrané uživatelské ID jako primární
toggle      přepnout mezi výpisem seznamu tajných a veřejných klíčů
pref        vypsat seznam předvoleb (pro experty)
showpref    vypsat seznam předvoleb (podrobně)
setpref     nastavit sadu preferencí pro vybrané uživatelské ID
keyserver   nastavit URL preferovaného server klíčů pro vybrané uživatelské ID
passwd      změnit heslo
trust       změnit důvěryhodnost vlastníka klíče
revsig      revokovat podpisu na vybraných uživatelských ID
revuid      revokovat vybrané uživatelské ID
revkey      revokovat klíč nebo vybrané podklíče
enable      nastavit klíč jako platný (enable)
disable     nastavit klíč jako neplatný (disable)
showphoto   ukázat vybrané fotografické ID
clean       odstranit nepoužitelné části z klíče
```

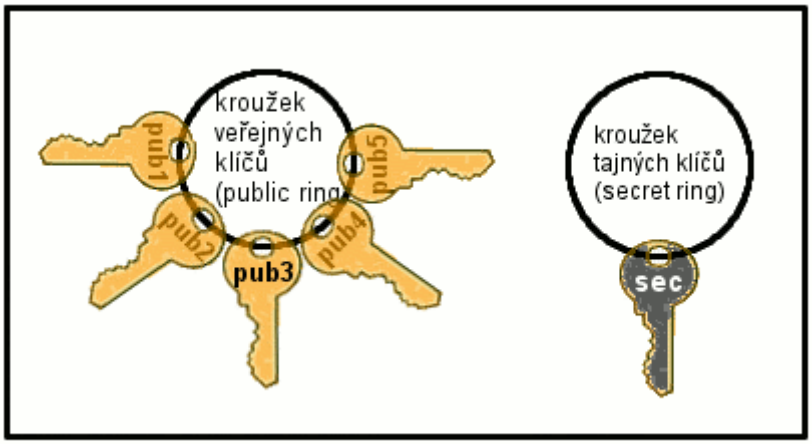
Nezapomeňte ukončit interaktivní režim příkazem **quit** nebo **save**.

## Zkratky se kterými se můžete setkat

```
pub = veřejný klíč (public key)
crt = X.509 certifikát
crs = X.509 certifikát a soukromý klíč přístupný
sub = podklíč - subkey
sec = tajný klíč (secret key)
ssb = tajný podklíč - secret subkey
uid = uživatelský identifikátor - fotka (user id)
uat = uživatelská vlastnost
sig = podpis (signature)
rev = zneplatňující podpis (revocation signature)
fpr = otisk prstu (fingerprint)
pkd = data veřejného klíče (public key data)
grp = rezervováno pro gpgsm (reserved for gpgsm)
rvk = zneplatňující klíč (revocation key)
tru = informace o databázi důvěry (trust database information)
spk = podpisový podbaliček
```

## ZM-SOFT formát GPG klíčů

GPG pracuje s kroužky na klíče! Má kroužek veřejných klíčů a kroužek s tajným klíčem.




**Kroužky na klíče GPG**

Nyní již víte, co znamená níže uvedený výpis.
sec 1024D/E43A40DF 2005-08-23 [platnost skončí: 2007-08-23]
pub 1024D/D1A4DF62 2006-01-07 [platnost skončí: 2008-01-07]

**Pro možnosti optimálního využití GPG klíčů forma ZM-SOFT zavedla následující formát:**

1. Klíč ID (ID): AE0B5090
2. Uživatel ID (UID): Zdeněk Maršík, Květná 1843, UHERSKÝ BROD
3. E-mail: marsik@zmsoft.cz
4. ČOP: 109777991 nebo EPZP: 80203111805172813666 (pro osoby mladší 15 let)
5. Fotka 91x114 pixelů (jpg)

Vysvětlivky:

- ✔ ČOP: číslo občanského průkazu
- ✔ EPZP: evropský průkaz zdravotního pojištění

Při **vytváření páru klíčů** pomocí příkazu `gpg --gen-key` by to mělo vypadat takto:

```
Jméno a příjmení: Zdeněk Maršík
E-mailová adresa: marsik@zmsoft.cz
Komentář: Květná 1843, UHERSKÝ BROD, ČOP:109777991
```

Po vytvoření páru klíčů **nezapomeňte přidat fotku,** velikost 91x114 pixelů ve formátu .jpg!

```
gpg --edit-key 68A4C546 <Enter>
addphoto <Enter> - přidání fotky
save <Enter> - uložení změn
```

## Podepisování veřejných klíčů

Podepisování slouží k ověřování pravosti klíčů. Veřejný klíč s vaším jménem si může vyrobit kdokoliv. Pokud si klíč předáte s nějakým známým osobně, budete si jist (a váš známý také), že jde o váš klíč. S každým partnerem se ale nebudete vždy setkávat osobně. Dejme tomu, že se znáte s Karlem a Bedřichem. Bedřichovi předáte váš klíč osobně a Bedřich váš klíč podepíše. Karel zná Bedřicha a získal od něho klíč. táhne si váš klíč ze serveru, ověří, že je podepsán Bedřichovým klíčem a má jistotu, že je to váš klíč.
Osobní setkání můžete také nahradit stažením ze serveru a telefonickým rozhovorem, ve kterém si vyměníte „otisk prstu“ (fingerprint).
Při budování důvěry v klíče mezi uživatele GNUPG nejde o pouhou stromovou strukturu, jako je tomu v případě X.509 certifikátu. Jde o zcela libovolný graf, říká se tomu „pavučina důvěry“ (orig. Web of Trust). Před nahráním na server veřejných klíčů by měl být váš veřejný klíč podepsán minimálně jednou jinou osobou.
Autoritativní podpisy pro GnuPG klíče vydává firma ZM-SOFT na občanský průkaz ( http://zmsoft.cz/ ).

## Nahrání vlastního veřejného klíče na server (jeho zveřejnění)

Nejprve si zjistíme, který je náš veřejný klíč příkazem:
`gpg -K`

```
C:\Documents and Settings\uzivatel>gpg -K
C:/Documents and Settings/uzivatel/Data aplikací/gnupg\secring.gpg
------------------------------------------------------------------
sec   1024D/E43A40DF 2005-08-23 [platnost skončí: 2007-08-23]
uid                  [jpeg image of size 3133]
uid                  Zdeněk Maršík (Květná 1843, UHERSKÝ BROD, ČOP:EK 166081) <marsikz@quick.cz>
ssb   4096g/62889E06 2005-08-23
```

Máme zde dvě možnosti:

1. Po vytvoření revokačního certifikátu můžeme uložit svůj veřejný klíč příkazem:
   `gpg --output D1A4DF62_zdenek_marsik.asc --export -a D1A4DF62`
   Pak si ho můžeme dát někým podepsat a vložit přes schránku na http://zmsoft.cz/gpg.html. Pak si ho může kdokoliv stáhnout a nahrát nebo ho podepsat a vystavit již podepsaný další osobou na serveru.


2. Odeslat svůj veřejný klíč přímo z kroužku veřejných klíčů příkazem:
   `gpg --recv-keys --keyserver pgp.mit.edu 0xD1A4DF62`

Vzhledem k nastavení FIREWALLů a různých ochran doporučuji používat první možnost!

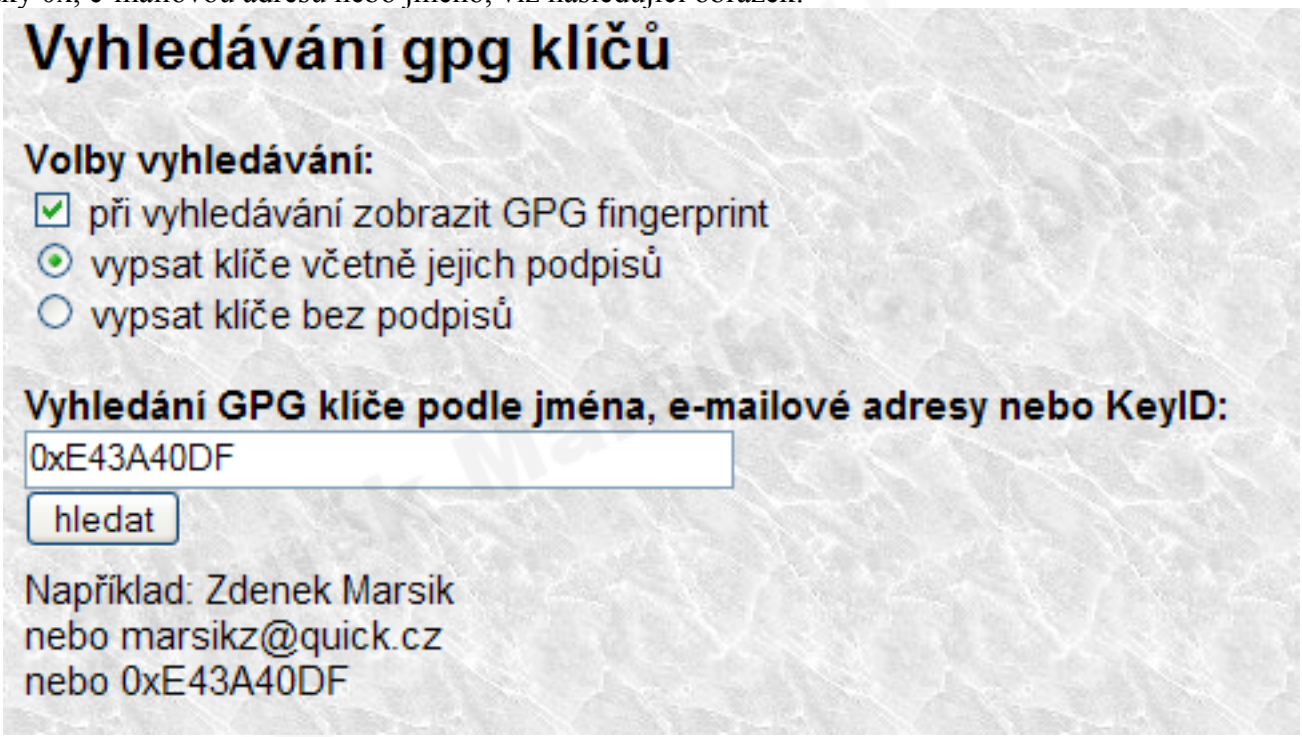
## Získání cizího veřejného klíče

a) na Internetu z http://zmsoft/gpg.html
b) v poštovním klientu THUNDERBIRD, OpenPGP, správa OpenPGP klíčů (doplněk ENIGMAIL)
c) přímo z Internetu: `gpg --recv-keys --keyserver pgp.mit.edu 0x974F5566`

Po načtení cizího veřejného klíče je třeba ověřit jeho FINGERPRINT. Buď danému člověku zavoláte a ověříte si FINGERPRINT nebo to ověříte na WEB stránkách toto člověka. Pokud se **FINGERPRINT shoduje,** tak **můžete cizí klíč podepsat** a začat používat. Nedoporučuji používat nepodepsané cizí klíče!

## Nalezení cizího veřejného klíče

Veřejné klíče lze hledat na http://zmsoft.cz/gpg.html. Pokud znáte ID, zadáte toto ID doplněné zepředu o znaky 0x, e-mailovou adresu nebo jméno, viz následující obrázek:




## Podepisování souborů:

I. `gpg -s soubor`
vytvoří podepsaný zkomprimovaný soubor s koncovkou .pgp který obsahuje i originální data

II. `gpg --clearsign soubor`
vytvoří podepsaný soubor s koncovkou .asc, která obsahuje i originální data a je v kódování 7 bitové ASCI, lze poslat emailem

III. `gpg -b soubor`
vytvoří soubor s příponou .sig, který obsahuje pouze binární formu podpisu

IV. `gpg -ba soubor`
vytvoří soubor s koncovkou .asc, který obsahuje pouze ASCII podobu podpisu

**Pro elektronickou fakturaci doporučuji používat III. gpg -b faktura123456.pdf. Vznikne soubor faktura123456.pdf.sig, což je elektronický podpis souboru faktura123456.pdf.**

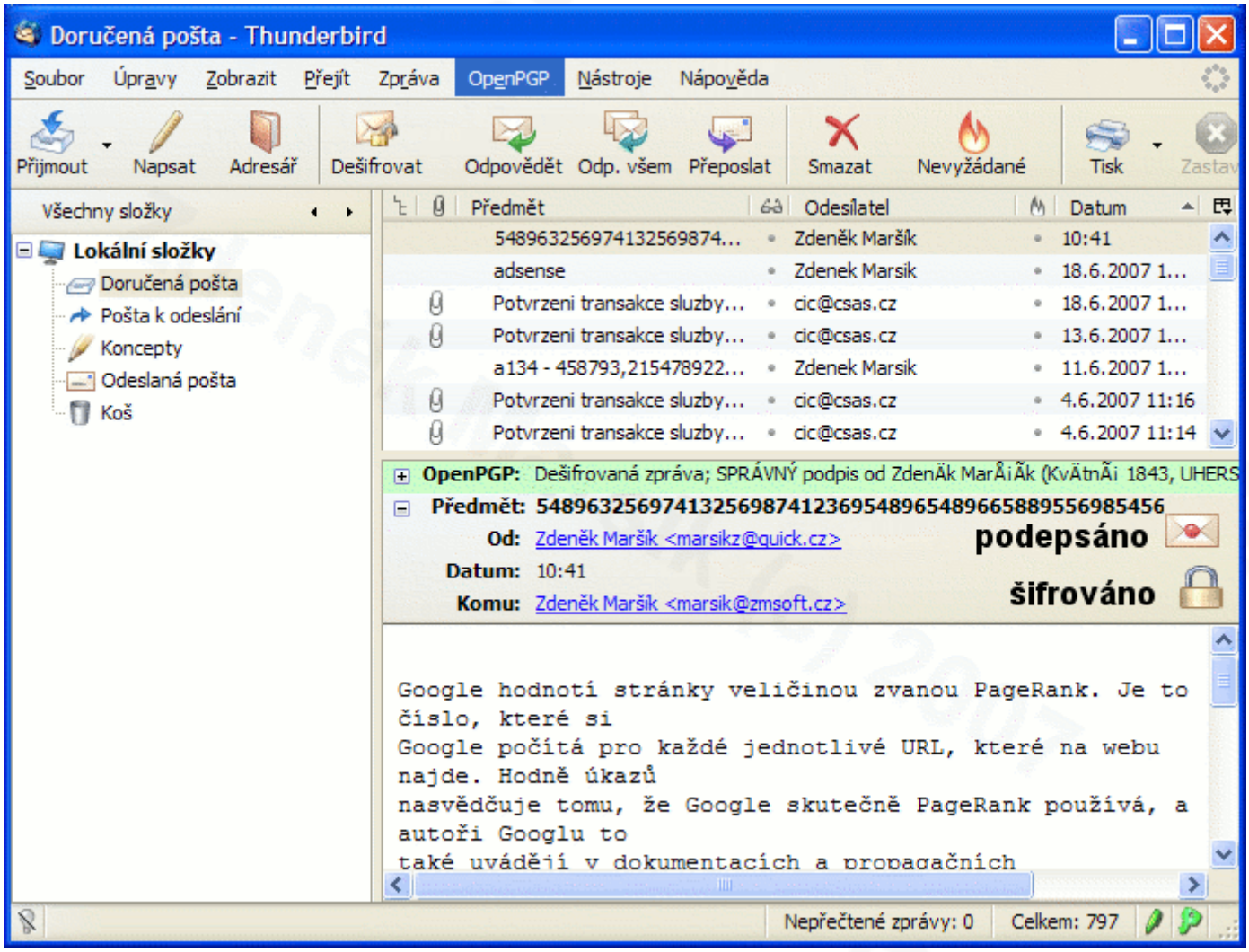
## II. THUNDERBIRD - poštovní klient

Poštovní klient THUNDERBIRD (hromopták) patří mezi jeden z nejlepších poštovních klientů. Má podporu UTF-8 psaní a čtení emailů, vynikající filtr nevyžádané pošty (spamu), možnost bezpečné komunikace a další vlastnosti, internetová adresa: http://thunderbird.czilla.cz/

Zde jsou vlastnosti poštovního klienta HROMOPTÁKA (thunderbirdu):

- Možnost zabezpečené komunikace
- Pokročilé nástroje pro správu pošty
- Podpora více účtů a identit
- Integrovaná čtečka kanálů
- Řada volitelných rozšíření a motivů
- Snadné ovládání v češtině
- Jednoduchá instalace
- Pro každého zdarma
- Účinná kontrola nevyžádané pošty (spamu)

Po instalaci a nastavení poštovního klienta THUNDERBIRD je potřeba nainstalovat třetí nástroj pro bezpečnou komunikaci, to je doplněk ENIGMAIL (určitě víte co to byla ENIGMA). Že je nainstalován poznáme podle nové nabídky v menu „OpenPGP“.



# III. ENIGMAIL - rozšíření pro THUNDERBIRD

Enigmail je rozšíření pro emailového klienta THUNDERBIRD o možnost šifrování, dešifrování, podepisování emailové komunikace a správu GnuPG klíčů. Enigmail se stahuje a instaluje jako doplněk poštovního klienta. Stáhnout lze z http://www.czilla.cz/doplnky/rozsireni/enigmail/

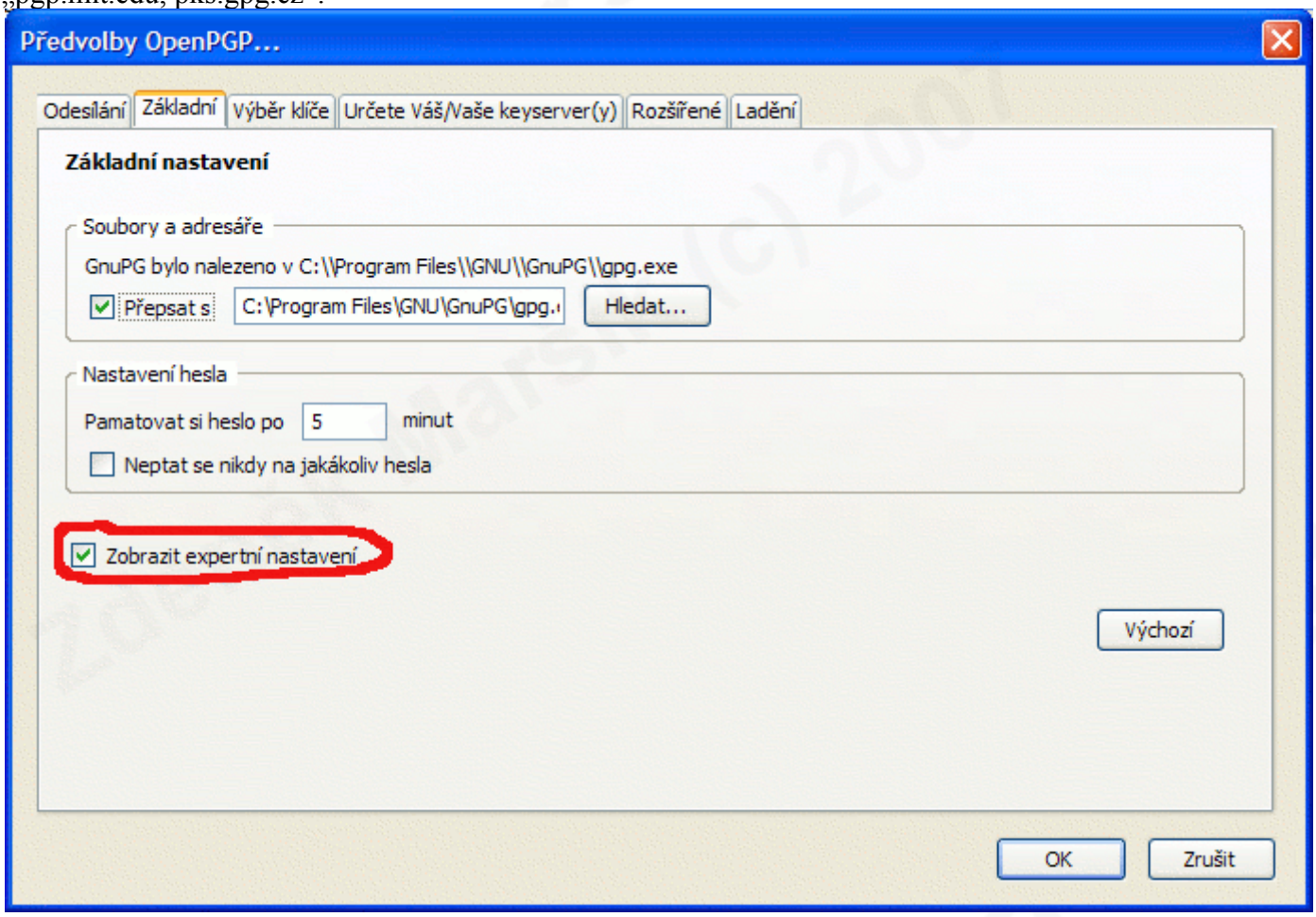
## Instalace doplňku enigmail

a) stažení z http://www.czilla.cz soubor enigmail-n.n.n.cs-CZ.thunderbird-linux.xpi
b) pomocí menu thunderbirdu nástroje, správce rozšíření, instalovat, vybrat soubor .xpi a nainstalovat

Po instalaci, ukončení a novém spuštění thunderbirdu se v menu objeví nová položka: OpenPGP
Pokud se neobjeví tak je něco špatně!

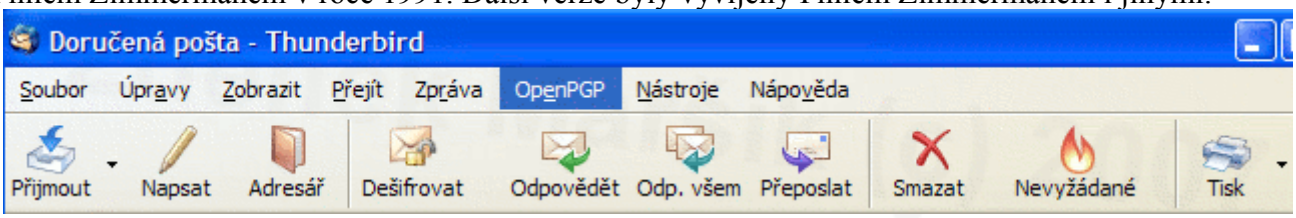
c) nastavení emailového účtu
V thunderbirdu (v LINUXu) menu, úpravy, nastavení účtu, openPGP/Enigmail
Zde zatrhnout „povolit podporu openPGP k této identitě (Enigmail), viz obrázek níže v „nastavení enigmailu“.

V tom samém menu ještě kliknout na tlačítko „rozšířené“, pak zatrhnout „zobrazit expertní zobrazení“ (viz obrázek níže) a v záložce „určete Vaše keyservery“ ponechat keyservery (servery veřejných klíčů) pouze „pgp.mit.edu, pks.gpg.cz“.



## OPENPGP - proč se enigmail nainstaluje jako OpenPGP

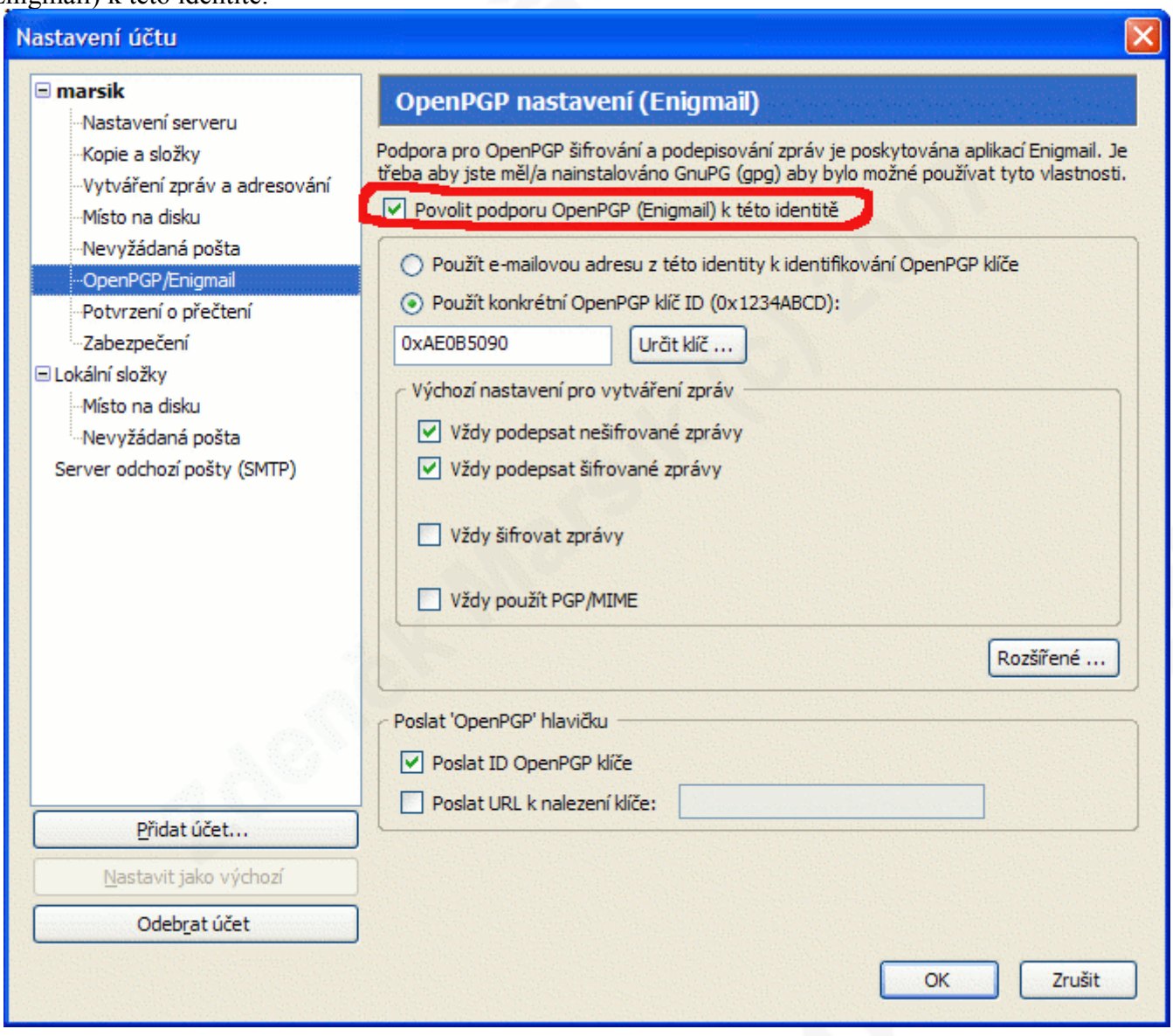
Pretty Good Privacy (PGP) (ang. dost dobré soukromí) je počítačový program, který umožňuje šifrování a podepisování. Je založeno na algoritmu RSA pro asymetrickou kryptografii. První verze PGP byla uvolněna Philem Zimmermanem v roce 1991. Další verze byly vyvíjeny Philem Zimmermanem i jinými.




PGP mělo takový vliv, že bylo standardizováno, aby byla umožněna spolupráce mezi různými verzemi PGP a podobného software. PGP bylo přijato jako internetový standard pod názvem OpenPGP. Nyní se jedná o otevřený standard dodržovaný PGP, GnuPG (GNU Privacy Guard nebo také GPG), Hushmail, Veridis, Authora a jinými.

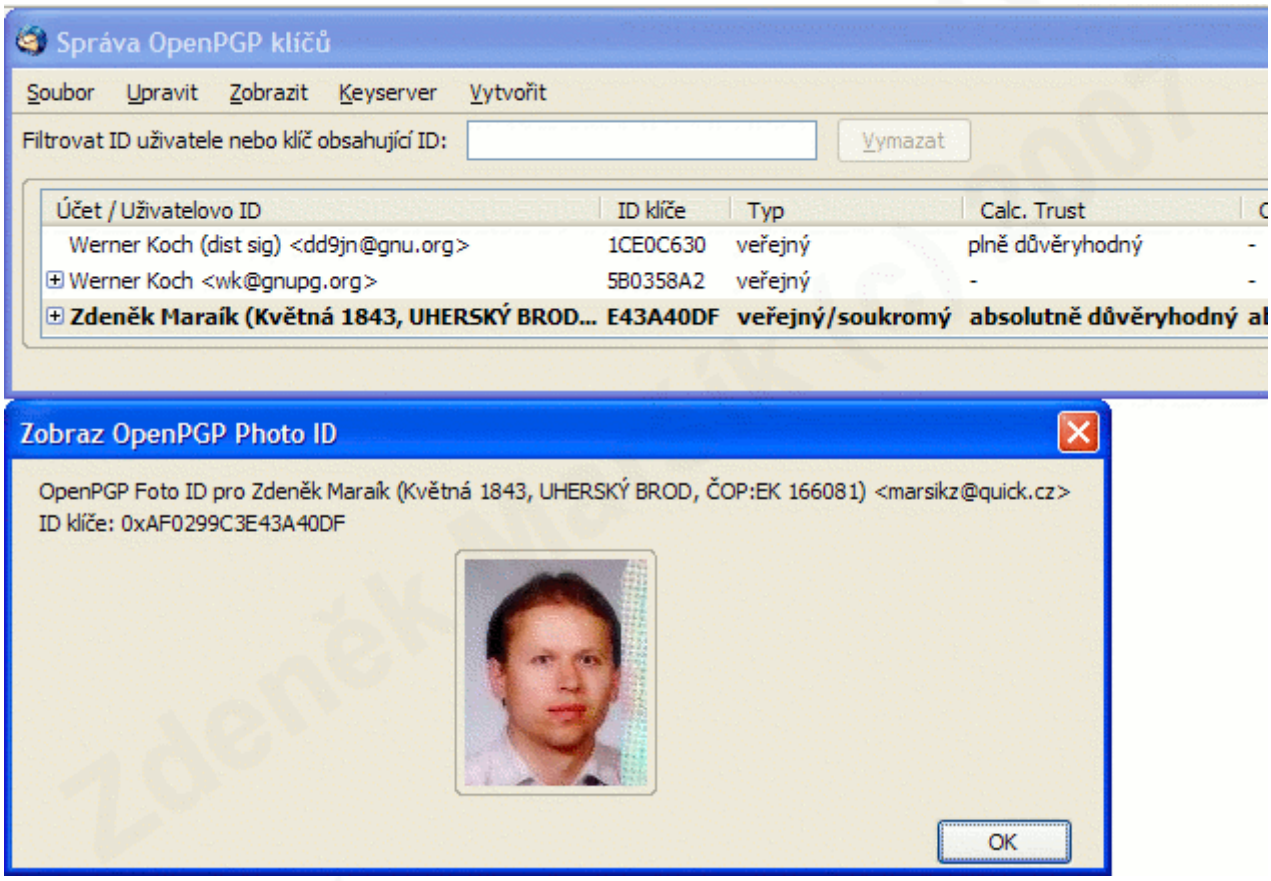
## Nastavení ENIGMAILu

V THUNDERBIRDu v menu nástroje, nastavení účtu, OpenPGP/Enigmail povolit podporu OpenPGP (Enigmail) k této identitě:



## Ovládání ENIGMAILu

Po instalaci, ukončení THUNDERBIRDu a jeho novém spuštění, lze již používat a ovládat gpg (GnuPG) přes enigmail v menu, OpenPGP, správa klíčů:

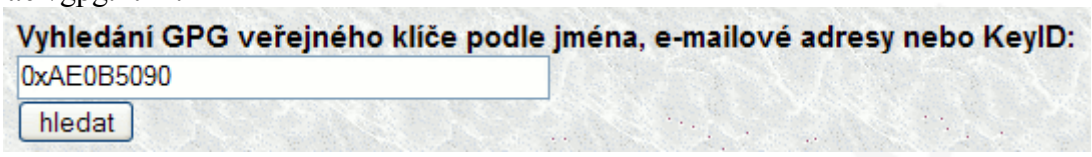



Jenom pozor při exportu veřejného klíče! Chybí mi v ENIGMAILu volba „exportovat veřejný klíč“. Zde je volba „odeslat veřejné klíče“ což odešle všechny veřejné klíče na vašem kroužku veřejných klíčů. Zde je nutno přejít do CONSOLY (příkazový řádek) a zadat:

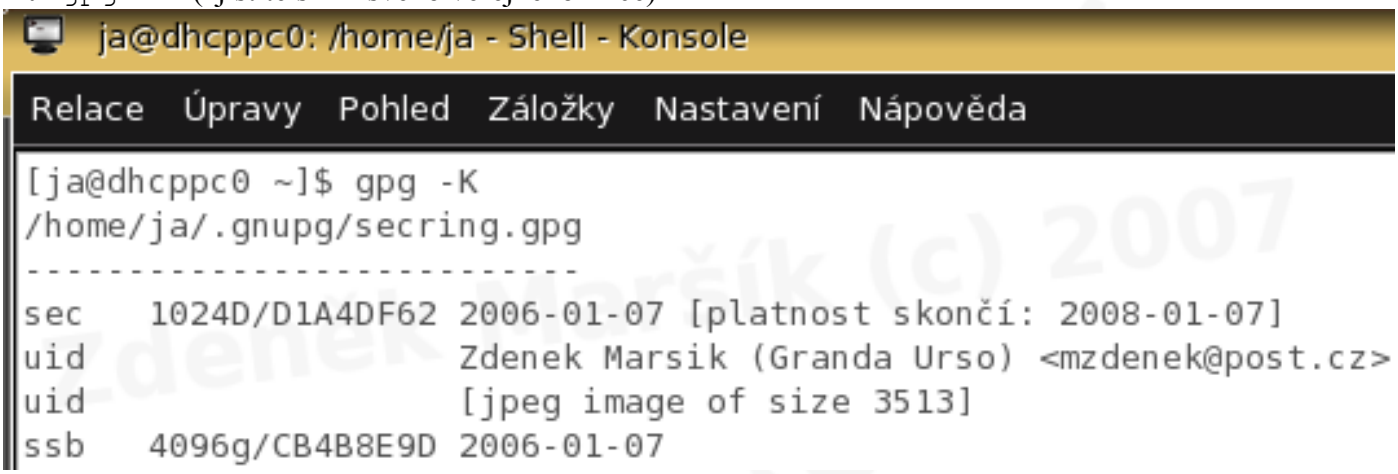
```
gpg -o zdenek_marsik.asc --export -a D1A4DF62
```

Dojde k vyexportování pouze jednoho veřejného klíče D1A4DF62 pod jménem zdenek_marsik.asc
Tento veřejný klíč je pak vhodné zaslat na server klíčů přes schránku WINDOWS nebo LINUXu na: http://zmsoft.cz/gpg.html

Upozornění: Neukládejte své veřejné klíče na vlastní webserver! Ztratíte vy i ostatní veřejnost kontrolu nad revokacemi veřejných klíčů. **Vždy ukládejte svoje veřejné klíče na server veřejných klíčů**,
viz http://zmsoft.cz/gpg.html
Pokud chcete **odkázat** na svůj veřejný klíč, tak to je nejlepší **odkazem na server veřejných klíčů**,
viz http://zmsoft.cz/sidlo_kontakty.html. Tedy konkrétně v mém případě je to odkaz na:
http://pgp.mit.edu:11371/pks/lookup?op=index&search=0xAE0B5090
Adresu tohoto odkazu najdete v **adresním řádku internetového prohlížeče** při nalezení veřejného klíče, na http://zmsoft.cz/gpg.html:

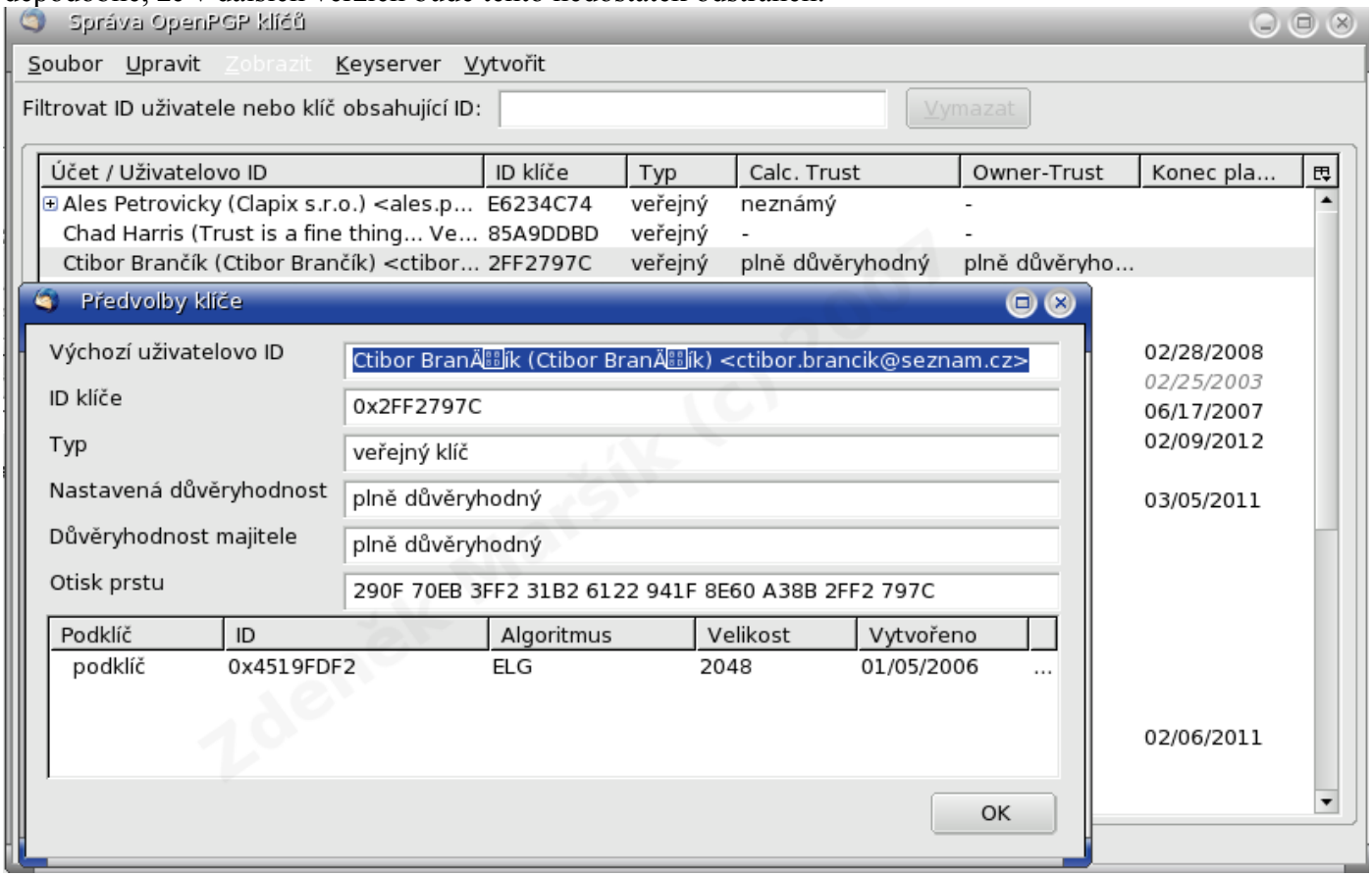


Možná pro vás bude jednodušší poslat tento klíč přímo z vašeho kroužku na klíče konzolovým příkazem:
`1. gpg -K` (zjistíte si ID svého veřejného klíče)




`2. gpg --send-keys --keyserver pgp.mit.edu 0xD1A4DF62` (Odeslání klíče, před ID se dává ještě 0x. Místo KeyID je možno zadat email nebo také jméno a příjmení.)

Tímto dojde k odeslání vašeho veřejného klíče na server pgp.mit.edu a jeho další rozeslání na další klíčové servery.
Nedostatkem doplňku ENIGMAIL je špatné zobrazování češtiny ve jménech s českými znaky. Je ale pravděpodobné, že v dalších verzích bude tento nedostatek odstraněn.




## Pozor!

**Je velmi důležité mít veřejný klíč na centrálním místě (pgp.mit.edu). Proto si nikdy svoje ani cizí veřejné klíče nevystavujte na web stránkách, pak by jste ztratili (a vaši partneři také) kontrolu nad revokacemi.**

# IV. Elektronický podpis v praxi

## Elektronický podpis

Princip je podobný jako v případě šifrování. Odesílatel připojí ke zprávě zvláštní dodatek, který se jednocestně vygeneruje z textu zprávy pomocí některé z hashovacích funkcí. Tento dodatek, tzv. „message digest“, zašifruje svým soukromým klíčem, čímž vznikne elektronický podpis. Příjemce si opatří veřejný klíč odesílatele, dešifruje přiložený podpis a porovná jej s tím, jak by měl vypadat podle jeho výpočtu. Pokud oba výsledky souhlasí, lze zprávu považovat za autentickou.

## Problematika e-podpisu

Zákon o elektronickém podpisu, je často bez podrobnější diskuze nad jeho praktickým uplatněním, prezentován jako mezní hranice, za kterou nás čeká báječný svět důvěryhodné elektronické komunikace. Ponechme stranou spekulace o podobě a obsahu prováděcí vyhlášky, která dle věcného záměru zákona určí konkrétní technický popis a podobu elektronického podpisu a technický postup při jeho ověření "ověřovatelem informací" a věnujme se krátkému popisu nejznámějších specifikací ověřování (často se spíše setkáme s pojmem certifikace ) digitálních podpisů.
Převážná většina současných systémů, které řeší problematiku ochrany integrity a autentizace přenášených dat, využívají služeb asymetrické kryptografie (někdy označované jako "kryptografie s veřejnými klíči"). Princip asymetrické kryptografie je založen na existenci páru klíčů pro každou komunikující entitu - veřejného a soukromého klíče. Soukromí klíč je důvěrný a každá komunikující entita se sama stará o ochranu jeho důvěrnosti a integrity. Naopak veřejný klíč je k dispozici volně celému komplexu (byť jen potenciálně) komunikujících stran. Vlastní mechanismus podpisu digitální zprávy pak zahrnuje: výpočet kontrolního součtu pro digitální zprávu (využitím vhodné hašovací funkce) a následné zašifrování výsledku soukromým šifrovacím klíčem. Příjemci zprávy pak stačí veřejným klíčem odesílatele dešifrovat kontrolní součet zprávy kterou obdržel, znova kontrolní součet zprávy vypočíst a porovnat oba výsledky. Jestliže se shodují, je v jisté úrovni akceptovatelného rizika zaručeno, že zprávu skutečně podepsal odesílatel a že zpráva nabyla během transportu změněna (pokud by první a/nebo druhý fakt byl nepravdivý, kontrolní součet zprávy, který příjemce obdržel od odesílatele a kontrolní součet nové příjemcem vypočtený by nesouhlasily).
Uvedený postup by mohl mylně vést k pocitu, že stačí zveřejnit seznam veřejných klíčů všech komunikujících stran a problematika ochrany integrity a autentizace přenášených dat je tímto vyřešena (samozřejmě za předpokladu, že výše uvedený postup používá matematických zákonitostí garantujících, alespoň v akceptovatelné úrovni rizika, nezpochybnitelnost výše uvedeného postupu, což v praxi existuje). Bohužel, otázka využití asymetrické kryptografie není takto triviální. Předpokládáme-li, že odesílatel nese plnou odpovědnost za ochranu důvěry a integrity svého soukromého klíče, stále ve výše uvedeném postupu schází mechanismus, kterým příjemce ověří, že veřejný klíč, který získal ze seznamu veřejných klíčů a použil jej k ověření integrity a autentizace přijaté zprávy, patří skutečně entitě, která klíč odeslala. K řešení tohoto problému se obvykle zavádí pojem "certifikační autorita" (Certification Authority (CA); v návrhu zákona se vyskytuje termín "ověřovatel informací"), nebo dokonce "důvěryhodná třetí strana" (Trusted Third Party). Certifikační autorita obvykle poskytuje infrastrukturu pro certifikaci veřejných klíčů uživatelů tak, aby byla umožněna spolehlivá komunikace mezi uživateli, kteří navzájem neznají své veřejné klíče, důvěryhodná třetí strana často nabízí i další služby - např. nepopiratelné časové razítka, spolehlivé služby elektronického notáře atp. Málokdy ovšem najdeme zmínku o problémech spojených s odvoláním (revokací) klíčů , resp. potřebou nezávislé revokační autority (Revocation Authority - RA) či využití znovu-potvrzení klíčů (rekonfirmace).

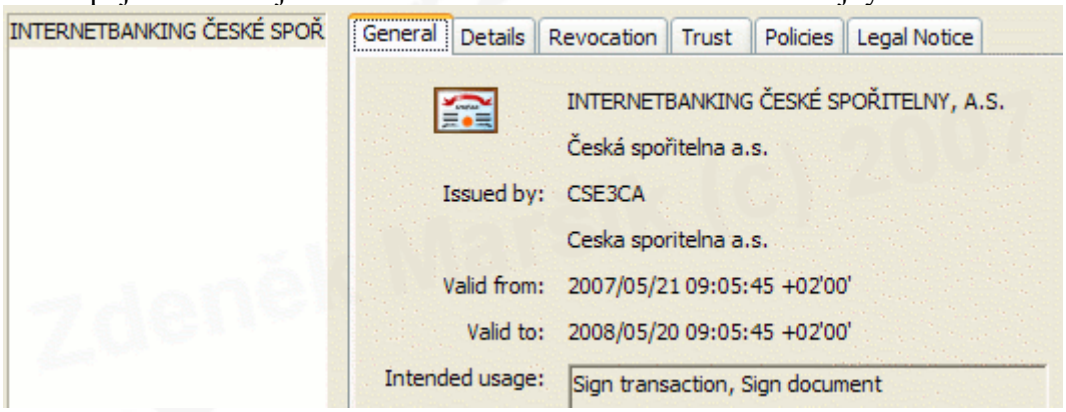
**Při kontrole klíčů lze v zásadě postupovat dvěma způsoby:**

1. **Konzervativním,** kdy je každý klíč, resp. certifikát považován za neplatný až do okamžiku, kdy jsme spolehlivým způsobem zpraveni o opaku, např. potvrzením vydaným CA spolu se spolehlivým označením času (timestamp).
2. **Liberálním,** kdy jsou certifikáty/klíče považovány za platné až do okamžiku, kdy jsme informování o opaku - např. prostřednictvím seznamu revokovaných certifikátů (CRL).

Podívejme se podrobněji na otázky certifikace pro dvě existující specifikace, které do určité míry ovlivňují většinu principů v implementovaných systémech. Jedná se o známé certifikáty X.509 a specifikace certifikace klíčů používaných Gnu Privacy Guard (GPG), i když za zmínku by jistě stála i poměrně nová specifikace (SDSI), která může v budoucnu hrát významnou roli nejen díky svým zajímavým rysům, ale také možnosti reálného nasazení (autoři Ron Rivest - MIT a RSA Data Security; Butler Lampson – Microsoft).

## Stromové struktury - X.509

Podoba certifikátů podle X.509 a s nimi související vztahy certifikace /revokace vzešly z návrhů pro globální databázi podle X.500. Podle X.500 mají všechny entity v celém světě odlišná jednoznačná jména - Distinguished Name (DN). Hierarchické uspořádání jmen přichází ruku v ruce s tímto návrhem tak, aby možné hodnoty (pojmenování) byly použitelné pro jakékoliv manipulace s daty. (Zde je na místě ještě dodatek "... podle představ plynoucích z jednoduchého pojetí relačních databází a také z (téměř) jednoznačných hierarchických struktur ve vládních organizačních jednotkách".) Pak se téměř samovolně nabízí jednoduché řešení spojit s každou jednoznačně identifikovatelnou entitou veřejný klíč - a máme X.509.

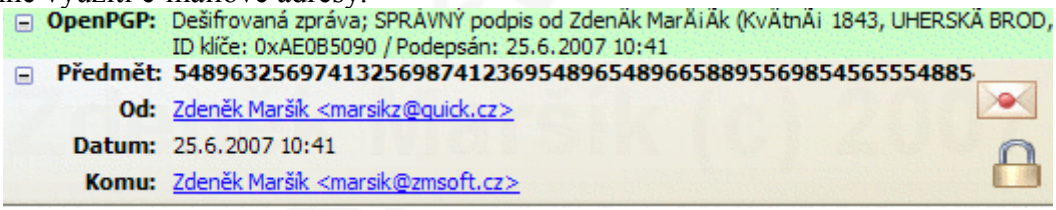



Takto sice není teoreticky omezena možnost DN spojené s více klíči, které praktické implementace to dnes ale umožňují? Významným rysem plynoucím z X.509 je stromová struktura, kdy daný uživatel a jeho klíč spadají pod jednu CA, ta pak může spadat pod další nadřazenou CA atd. až se dostaneme ke kořenové CA. Problém je, že celý svět nefunguje podle jednoznačných hierarchických struktur a kdyby náhodou fungoval, tak není jisté, zda by bylo možné jej "zorganizovat" podle X.500 bez jakýchkoliv přílišných zjednodušení a záměn. Další problém je v tom, že jednoznačné pojmenování existuje (snad) jen v určité doméně, např. organizaci. V této doméně lze s určitým úsilím a v závislosti na její velikosti také prosadit jednoznačnou bezpečnostní politiku a zavést určité konvence. Rozhodně ale nelze očekávat, že i další domény, jejichž činitelé se budou zúčastňovat výměny informací, budou mít jednoznačnou bezpečnostní politiku nebo dokonce politiku obdobnou až totožnou.
Certifikáty X.509 jsou vždy vytvářeny certifikační autoritou, která je také revokuje. Zde je další problém - CA může velmi jednoduše ovlivnit a falzifikovat podstatná data v neprospěch strany, jejíž pár kryptografických klíčů je předmětem sporu. Podle X.509 je totiž certifikační autorita zároveň i revokační autoritou a navíc je struktura seznamu revokovaných certifikátů (Certificate Revocation List – CRL) certifikační autoritou libovolně modifikovatelná aniž by modifikace byla bezproblémově a jednoznačně prokazatelná. Pro řešení tohoto problému je potřeba jednak jednoznačně oddělit RA od CA, ale také zabránit triviální modifikaci CRL (např. řetězením hodnot získaných aplikací kryptografických hašovacích funkcí na postupně publikované CRL). I samotná rychlost a rozsah zpřístupnění CRL je problémem, který není dosud dostatečně analyzován a uspokojivě řešen v praktických implementacích.

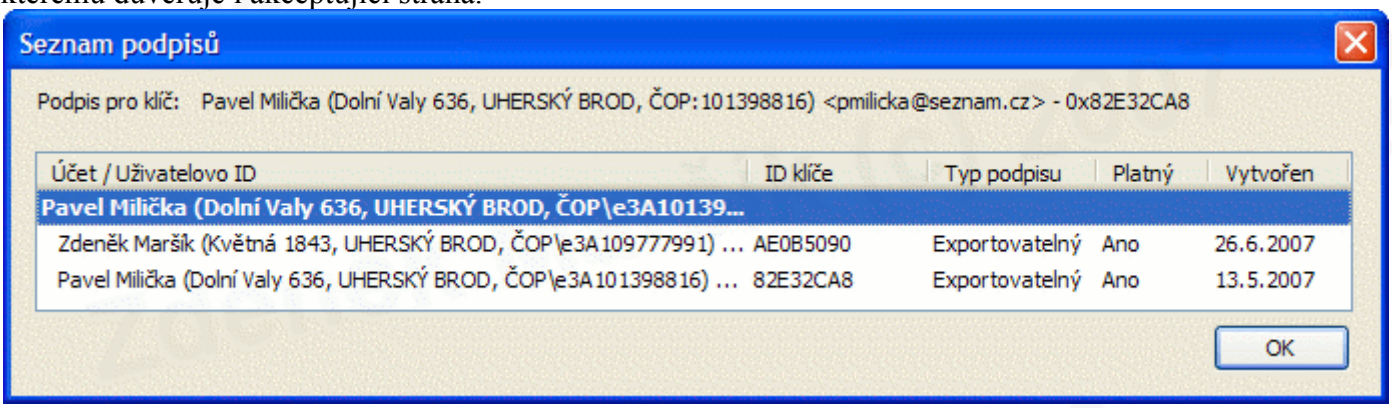
Pro ověření certifikátu X.509 je potřeba mít k dispozici také veřejné klíče (certifikáty) všech nadřízených CA až po tu CA, které lze bezmezně důvěřovat a jejíž klíč je "zaručeně" bezpečný. Toto obvykle znamená potvrzení nebo dodání certifikovaného klíče nezávislou cestou (nejlépe zveřejnění v tisku, osobní odběr od důvěryhodného zástupce CA atd.). V žádném případě nelze za spolehlivou metodu považovat zveřejnění klíče na Internetu, jeho dodání v rámci aplikace (Netscape Navigator) apod. Navíc, jakýkoliv podvod nebo selhání certifikační autority znamená porušení důvěry ve všech větvích stromu certifikačních vztahů pod touto autoritou. (Pozn. nemusí se ani jednat o její podvod nebo selhání - velmi účinné útoky typu odmítnutí služby lze způsobit ovlivněním jí nadřízené CA a revokací odpovídajícího páru klíčů .)
V praxi se s implementací X.509 lze nejčastěji setkat v populárních prohlížečích Thunderbird aj. Vlastní implementace má však několik vážných nedostatků. K nejvážnějším nedostatkům patří výrazné omezení délky šifry, se kterou pracují použité šifrovací algoritmy od Microsoftu- a tím i síly šifrování, díky čemuž je standardně dodávaná implementace mimo rozsah reálného použití.

## Pavučina důvěry, standard IETF RFC 2440 - GnuPG

**Gnu Privacy Guard** a právě její způsob spojení páru kryptografických klíčů s určitým uživatelským identifikátorem (UserID) je vážným konkurentem ovlivňujícím rychlost i rozsah implementací systémů na bázi X.509. Klasickou konvencí GPG při spojení klíčů s UserID je ovšem to, že UserID se skládá z uživatelského jména a e-mailové adresy. Tak jako je struktura X.509 do určité míry jednoznačnou díky DN, tak je pro GPG významné využití e-mailové adresy.




Významným rysem certifikace GPG klíčů je fakt, že tuto certifikaci si provádí každý činitel na základě vlastních informací a podle vlastního rozhodnutí (lze ovšem umožnit i zprostředkované "představení" neznámých klíčů = činitelů). Pak spočívá management důvěry přímo pod kontrolou uživatele. Velkou nevýhodou obvyklého využití GPG klíčů je fakt, že není jasně prosazována homogenní bezpečnostní politika ani uvnitř jedné domény a organizace (doména) spoléhá v relevantních kritických funkcích na korektní a zodpovědný přístup svých členů.
Problémy ohledně akceptování cizích klíčů GPG lze do určité míry řešit vazbou na požadavek mít klíč podepsán více (nezávislými) činiteli, v tomto případě ale nelze s jistotou spoléhat na to, že řetěz podpisů (klíč A je podepsán klíčem B, který je zase podepsán klíčem C atd.) obsahuje podpis (certifikaci) činitelem, kterému důvěřuje i akceptující strana.




Jelikož "pavučina důvěry" je obecnější datovou strukturou než struktura stromová, typická pro X.509, není nijak vyloučeno použít GPG a vybudovat na jejím základě konstrukci podobnou hierarchickým strukturám X.509. Novinkou je také akceptování klíčů ve formátu X.509. Z praktického hlediska se s GPG lze setkat v podobě samostatného softwarového balíku GnuPG (PGP se stalo plně komerční aplikací). Jedná se o samostatný aplikační balík, který zahrnuje prostředky pro generování páru klíčů, komunikaci s

certifikačním serverem (serverem veřejných klíčů) a prostředky pro šifrování a digitální podepisování elektronické pošty a souborů. **GPG využívá výhradně dlouhé šifry, jelikož nepodléhá regulaci vývozu ze země svého původu.** Zároveň velmi pečlivě nakládá se soukromým klíčem, který je uchováván na disku či v operační paměti - do jisté míry tak řeší problematiku správy veřejných klíčů popsanou níže. Velké popularitě GnuPG jistě přispívá i dostupnost zdrojových kódů jednotlivých implementací.

## Obecné problémy spojené se systémem veřejných klíčů

Zajímavá (a často opomíjená) je skutečnost, že asymetrická kryptografie má z globálního pohledu malé náklady na infrastrukturu proto, že uživatelé systémů s veřejnými klíči nesou značné skryté "administrativní" zatížení. Systém veřejných klíčů totiž spoléhá na uživatele v tom, že opravdu zodpovědně a řádně kontrolují veřejné klíče svých partnerů v komunikaci a také že bezpečně spravují své soukromé klíče. Obě skupiny úloh jsou netriviální, systémy s veřejnými klíči ovšem v naprosté většině dnešních implementací postrádají spolehlivou jednotnou infrastrukturu k jejich prosazování. Je zmíněno pět zásadních problémů spojených se systémy veřejných klíčů :

1. Autentizace uživatele - jak CA autentizuje vzdáleného uživatele při vydání prvotního certifikátu veřejného klíče?
2. Autentizace CA - asymetrická kryptografie sama o sobě nemůže zabezpečit distribuci a ověření veřejného klíče vrcholné (kořenové) CA.
3. Seznam revokovaných certifikátů (CRL) - včasná a bezpečná revokace představuje neuvěřitelný problém ohledně rozšiřitelnosti a výkonu. Díky tomu je také využití systémů s veřejnými klíči (často) bez patřičné revokační infrastruktury.
4. Správa soukromých klíčů - uživatel má svůj dlouhodobý tajný klíč v paměti počítače při přihlášení (ale často i při dalších operacích, nikoliv výjimečně v cache paměti i po celou dobu práce se systémem).
5. Kvalita vstupní fráze - není způsob jak skutečně přimět uživatele k výběru kvalitní vstupní fráze (tento problém je dobře znám u systémových a aplikačních hesel).

Zatímco problémy 4 a 5 jsou společné i pro systémy symetrické kryptografie, první tři problémy jsou netriviálními záležitostmi, často ponechávané "mimo diskuzi" i v jinak hodnotných akademických příspěvcích či dokonce zamlžovány v konzultantských zprávách pro design bezpečnostní architektury. Nemůže být pochyb o tom, že otázka certifikace a revokace klíčů je zásadním problémem při zvažování struktury bezpečnostní architektury a jejího zapojení do (stávajícího) IS či celé organizace tak, aby management důvěry v elektronické podobě skutečně odpovídal vztahům důvěry a organizačním souvislostem v reálné podobě. Toto je totiž naprosto zásadní princip pro implementaci pokročilých bezpečnostních opatření.

# V. DODATKY

## Symetrická kryptografie

je založena na tom, že oba účastníci komunikace na nechráněných kanálech používají stejný klíč. Problémem je, jak bezpečně doručit stejný klíč druhému účastníkovi komunikace. Zde se používá Diffie-Helmannův algoritmus výměny exponenciálního klíče.

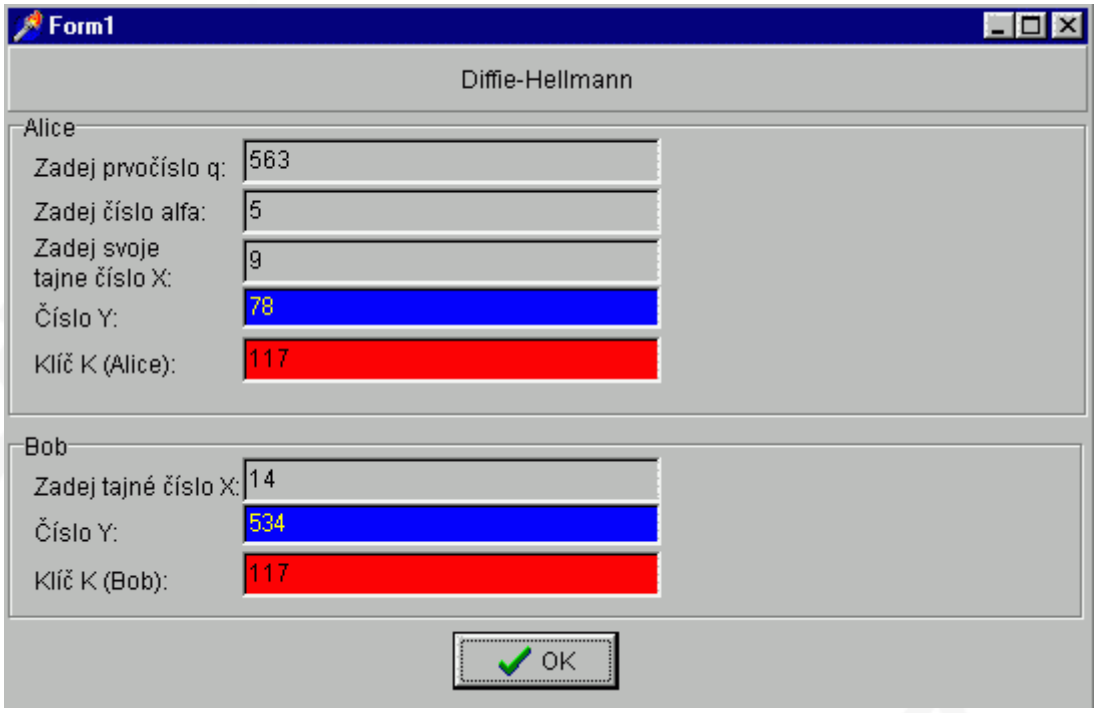
## Diffie-Hellmanův algoritmus výměny symetrického klíče

********** Diffie - Hellman **********
Dříve, než spolu začnou dva lidé komunikovat, musí se dohodnout na velkém prvočísle Q a na čísle @ (alfa), kde @ je primitivní modulo mod Q. Čísla @ a Q může znát každý, nejsou tajná. Jeden člověk, řekněme Alice, vybere náhodné číslo X(a). Toto číslo uloží jako tajné a podle následujícího vzorce vypočítá číslo Y(a):

Ya=@^Xa mod Q
Alice toto výsledné číslo odešle svému partneru Bobovi. Mezitím si Bob vybral své vlastní náhodné číslo Xb a odpovídající číslo Yb poslal Alici:
Yb=@^Xb mod Q
Alice teď vypočítá číslo Kab, jejich sdílený klíč, a to podle následujícího vzorce:
Kab=Yb^Xa mod Q
Mezitím si Bob vypočítá to stejné číslo Kab a to podle vzorce:
Kab=Ya^Xb mod Q
Číslo Kab je bezpečné, protože jeho výpočet vyžaduje, aby jste znali čísla Xa, Xb. Alice zná číslo Xa a Bob zná číslo Xb. Ten, kdo by odposlouchával jejich linku, by znal pouze čísla Ya, Yb.Vypočítat číslo Xa z čísla Ya nebo Xb z čísla Yb je velmi obtížný problém.
Zkusme si tento příklad předvést na reálných číslech. Řekněme, že jsme se rozhodli použít tato čísla:
@=5
Q=563
Alice si vybrala číslo 9
Xa=9
Potom Bobovi odeslala číslo 78
Ya= 5^9 mod 563 = 78

Bob si vybral číslo 14
Xb=14
Alici pak poslal číslo 534
Yb=5^14 mod 563 = 534
Alice pak vypočítala číslo Kab=117
Kab=534^9 mod 563 = 117

Bob vypočítal číslo Kab=117
Kab=78^14 mod 563 = 117




Alice a Bob teď spolu mohou komunikovat pomocí šifrovacího klíče 117.
********** END Diffie - Hellman **********
Příklad diffie-hellmanova výpočtu:


Vybrané prvočíslo Q=33579349
Vybrané číslo @=11
Tajné číslo Alice Xa=13

Ya=@^Xa mod Q
Ya=11^13 mod 33579349 = 34522712143931 mod 33579349 = 18492474 (tedy 34522712143931 : 33579349 = 1028093 a zbytek po dělení je 18492474)
Ya=18492474
Yb=@^Xb mod Q
Yb=11^7 mod 33579349 = 19487171 mod 33579349 = 19487171 (tedy 19487171 : 33579349 = 0 a zbytek po dělení je 19487171)
Yb=19487171
Kab=Yb^Xa mod Q
Kab=19487171^13 mod 33579349 =
584432487303310164233763622201135452426129804252662127149218322169937620111196422505375923954 11 mod 33579349 = 17971067
Kab= 17971067

**Alice má nyní symetrický klíč 179771067.**

Tajné číslo Boba Xb=7

Yb=@^Xb mod Q
Yb=11^7 mod 33579349 = 19487171 mod 33579349 = 19487171 (tedy 19487171 : 33579349 = 0 a zbytek po dělení je 19487171)
Yb=19487171
Kab=Ya^Xb mod Q
Kab=18492474^7 = 739545872170936885676040832101517220417381481949824 mod 33579349 = 17971067
Kab=17971067

**Bob má nyní symetrický klíč 17971067.**

# VI. ZÁVĚR

## Využití GnuPG

GnuPG používá veřejné klíče dle standardu IETF RFC 2440 a k ukládání veřejných klíčů používá centrální distributovanou databázi. To znamená, že nahrání jednoho veřejného klíče na libovolný GPG server k tomu určený, se projeví na všech ostatních GPG serverech veřejných klíčů. Tím jsou vyřešeny problémy se změnami a odvoláváním (revokacemi) veřejných klíčů. Navíc firma ZM-SOFT zavedla jistá pravidla a provádí autoritativní podpisy. Tím je autentičnost veřejných klíčů zvýšena na úroveň X.509 certifikátů. Navíc ještě umocněná pavučinou důvěry, která zůstala zachována.
GnuPG – GPG lze použít v podnikatelské sféře, kdy již nikdo nemůže zpochybňovat Vaše objednávky, elektronické faktury apod. Navíc lze použít **nejsilnější zabezpečení = šifrování + digitální podpis**, které je prakticky neprolomitelné a bezpečnost takových zpráv je na nejvyšší úrovni.