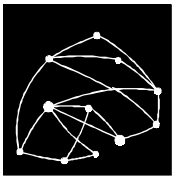
www.projetoderedes.kit.net



RNP
REDE NACIONAL DE PESQUISA

Curso Introdutório ao uso do PGP.

Fábio Rogério Hideki Okamura

6 de maio de 1998

Documento N° RNP / REF / 0181
VERSÃO PRELIMINAR

Sumário

Resumo

Trata-se de um curso básico destinado a quem não têm conhecimento algum sobre o uso do PGP. Quer seja como aplicação ´standalone´, quer seja em conjunto aos clientes de e-mail mais comumente utilizados pelo Núcleo de Apoio da RNP de Campinas (NA-CP): ´Eudora´ e ´Pine´.

1. Introdução

Do grego:

Krypto=oculto, escondido

Grapho=escrita, grafia

Temos:

Criptografia="A arte de tornar incompreensível, com observância de normas especiais consignadas numa cifra de código, o texto de uma mensagem escrita com clareza" (Dicionário Aurélio).

"O e-mail é seu. É particular, privativo e ninguém tem o direito de bisbilhotar. PGP dá ao usuário o poder da privacidade" (Guia de Referência Rápida PGP).

PGP (Pretty Good Privacy) é um software de criptografia multiplataforma de alta segurança produzido pela empresa Pretty Good Software agora fundida à Networks Associates Inc (http://www.nai.com).

Seus pontos fortes são:

- tecnologia de chaves públicas, logo não é necessário nenhum canal seguro para troca de chaves
- somente a pessoa a quem é destinada o email pode lê-lo
- autenticação de remetentes a fim de verificar se quem aparece como remetente foi que enviou a mensagem

2. Como PGP funciona

2.1 Tecnologia de chaves públicas

Encriptação convencional, também chamada de encriptação simétrica ou encriptação por chave-única, era o único tipo de encriptação usado antes da introdução do método da chave-pública no final de 1970. E, de longe, ainda permanece o esquema mais utilizando entre os dois tipos de encriptação. Podemos imaginá-lo, como uma caixa com uma tranca. Nela colocaríamos os documentos a serem transferidos. Tanto o remetente como o destinatário detém uma cópia da chave. O problema é como transferirmos a cópia da chave ao remetente. Pois a chave pode se extraviar, ou ningém pode garantir que o envolope com a chave tenha sido violado e o segredo copiado. Eis porque dizemos que no esquema de encriptação convencional necessitamos de um canal seguro para troca das chaves antes de iniciarmos a troca dos documentos encriptados.

Pode-se dizer que o esquema de chaves-públicas, proposto em 1976 por Diffie e Hellman, foi a primeira verdadeira revolução na área de encriptação em centenas de anos. Desta vez os algoritmos das chaves são funções matemáticas e não simples operações nos padrões de bits. Neste método, temos duas e não uma única chave como na encriptação simétrica. Uma analogia seria uma caixa que possui uma tranca de três posições, sendo a posição central aberta e tanto a esquerda como direita fechada; e duas chaves: uma pessoal (privada) que só gira a tranca para esquerda e outra pública (para livre distribuição) que gira a tranca para direita. Tudo que temos que fazer é distribuir a chave pública para nossos possíveis

remetentes. Eles trancam a caixa para direita e somente o destinatário (o que possui a chave privada) pode destrancar (virando a tranca para esquerda). Vale notar que tudo que for trancado por uma chave pode ser aberto por outra e vice-versa - uma importante característica deste esquema. Outra característica é inviabilidade, computacionalmente falando, de se descobrir a chave de decriptação conhecendo o algoritmo e a chave de encriptação.

Mas nem tudo são rosas. Comparado com a encriptação convencional, o algoritmo de encriptação por chave-pública é por demais lento. Então por que o PGP utiliza o método? Porque o esquema oferece uma enorme flexibilidade para realizar operações de segurança como manuseio de chave e autenticação.

2.2 PGP agilizando a encriptação

Se o algoritmo de encriptação por chave-pública é muitíssimo lento, então deve-se utilizá-lo somente para proteger pequena quantidade de dados.
PGP agiliza a encriptação lançando mão do método de chave-única. Ao receber o texto, PGP primeiro realiza a compressão do mesmo (via algoritmo ZIP), depois gera um número aleatório relativo àquela sessão somente (logo, cada encriptação de um mesmo documento gera saídas diferentes); esse número é usado como chave para encriptar o documento comprimido via encriptação convencional. É essa a chave a ser encriptada usando o método de chaves-públicas. PGP utiliza a chave pública do destinatário da mensagem para encriptá-la. Como somente o destinatário detém sua respectiva chave privada, só ele pode recuperar a chave gerada pela sessão de encriptação e com ela recuperar o texto comprimido, descomprimindo-o em seguida; obtendo dessa forma o documento original. Eis porque dissemos que o método de chave-pública oferece grande flexibilidade no manuseio de chaves.

2.3 PGP autenticando mensagens

PGP também pode ser utilizado para autenticar mensagens.

Antes de comprimir o texto, PGP utiliza uma função de espalhamento segura no conteúdo do texto. Essa função retorna uma identificação única a mensagem; identificação esta que muda, caso o texto sofra qualquer alteração de conteúdo. Podemos fazer uma analogia a uma função de somatório sofisticada (uma espécie de CRC da mensagem). Esse processo é conhecido como assinatura digital. Essa identificação é então encriptada utilizando a chave secreta do remetente e depois anexada a mensagem. Quando do recebimento, PGP descomprime e então usa a função sobre a mensagem que recebera. Depois tenta extrair a identificação encriptada anexa a mensagem utilizando a chave pública do remetente a fim de identificar se realmente foi o remetente que enviou. Depois compara as identificações a fim de verificar se o texto foi adulterado.

Em resumo:

- Extrai um identificador *I*, de 128 bits, único a mensagem, via algoritmo MD5.
- Comprime a mensagem via ZIP.
- Gera uma chave *K* aleatória válida somente para sessão corrente de encriptação.
- Encripta a mensagem via encriptação simétrica, utilizando a chave *K* do passo

anterior. Algoritmo utilizado é o CAST ou IDEA ou em menor grau o 3DES e a chave tem 128 bits de tamanho.
- Encripta a chave *K* utilizando a chave pública do destinatário via algoritmo RSA ou DH/DSS (768 bits de tamanho no mínimo). Anexa o resultado a mensagem
- Encripta *I* usando a chave secreta do remetente. Anexa o resultado a mensagem.
- Ao receber, PGP recupera *K* usando a chave privada do destinatário.
- Desencripta a mensagem com *K*.
- Descomprime o texto.
- Extrai *I'* (identificador da mensagem recebida) via MD5.
- Obtém *I* a partir da chave pública do remetente.
- Compara *I* com *I'*.

3. Usando o PGP

3.1 Gerando seu próprio par de chaves

O primeiro passo no uso do PGP é a criação do par de chaves (pública e privada) do usuário.

Vamos mostrar como fazê-lo nos três ambientes mais comumente utilizados pelo Núcleo da Rede Nacional de Pesquisa de Campinas: DOS, Windows e Unix.
O Núcleo de Campinas utiliza a versão 2.6.Xi no DOS e no UNIX e versão 5.Xi no Windows e Unix.

3.1.1 Via PGP 2.6.3i (DOS/UNIX)

Para DOS e Unix temos uma etapa anterior a geração de chaves que denominaremos de preparação.

A preparação para DOS é simplesmente adicionar as seguintes linhas no arquivo AUTOEXEC.BAT, incluir o PGP no PATH do sistema e reiniciar o micro.

```
SET PGPPATH=C:\PGP
SET TZ=GMT+3
```

O primeiro parâmetro informa onde se encontra instalado o PGP, de forma a que este possa encontrar a base de dados de chaves, denominada pelo programa de Anel de Chaves (KeyRing, chaveiro). Temos dois Anéis: um para chaves públicas (PubRing) e outras para chaves privadas (SecRing).

O segundo parâmetro é utilizado no PGP para lidar com TIMESTAMPS.

Existem muitos outros parâmetros que permitem customizar ainda mais o comportamento do PGP. Vale a pena checar o Guia do Usuário que acompanha o software; o mesmo vale para o Unix.

A preparação do Unix consiste em criar um diretório `.pgp` e modificar as permissões desse de forma a somente o usuário proprietário das chaves possa ter acesso ao mesmo.

> **cd ; mkdir .pgp ; chown 700 .pgp**

Para gerar nosso par de chaves no PGP v2.6.Xi (DOS e Unix) utilizamos:

> **pgp -kg**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/04/27 13:54 GMT

Pick your RSA key size:
1) 512 bits- Low commercial grade, fast but less secure
2) 768 bits- High commercial grade, medium speed, good security
3) 1024 bits- "Military" grade, slow, highest security
Choose 1, 2, or 3, or enter desired number of bits:

Agora entramos com o tamanho da chave (*1* para *512bits*, *2* para *768bits* e *3* para *1024bits*). Quanto maior o número de bits maior a segurança e mais demorada é a operação de geração.

Choose 1, 2, or 3, or enter desired number of bits: **2**

Generating an RSA key with a 768-bit modulus.

You need a user ID for your public key. The desired form for this
user ID is your name, followed by your E-mail address enclosed in
<angle brackets>, if you have an E-mail address.
For example: John Q. Smith <12345.6789@compuserve.com>
Enter a user ID for your public key:

Nesse passo o PGP pede a identificação e endereço eletrônico do proprietário do par de chaves, preferencialmente no formato que o PGP exemplifica.

Enter a user ID for your public key:
Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>

You need a pass phrase to protect your RSA secret key.
Your pass phrase can be any sentence or phrase and may have many
words, spaces, punctuation, or any other printable characters.

Enter pass phrase:

O passo seguinte é fornecer uma frase segredo. Nessa frase será aplicado o algoritmo MD5 gerando um ID de 128 bits que será a chave de encriptação (por chave-única) do anel secreto (SecRing). Isso impossibilita que intrusos que venham a ter acesso ao seu micro/sua área obtenham sua chave secreta.

Enter pass phrase:
Enter same pass phrase again:
Note that key generation is a lengthy process.

We need to generate 259 random bits. This is done by measuring the time intervals between your keystrokes. Please enter some random text on your keyboard until you hear the beep:
259

Agora o PGP pede que o usuário digite teclas aleatoriamente. A taxa de digitação será usada como base para geração de números aleatórios usados pelo PGP.

0 * -Enough, thank you.
..**** .....****
Pass phrase is good. Just a moment....
Key signature certificate added.
Key generation completed.

A chave foi gerada! Para visualizá-la entre com: *pgp -kv* ou *pgp -kc* (versão completa).

> **pgp -kv**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/04/27 14:44 GMT

Key ring: 'c:\pgp\pubring.pgp'
Type Bits/KeyID Date User ID
pub 768/9EC7BF9D 1998/04/27 Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
1 matching key found.

Para visualizar com o 'fingerprint' (a assinatura MD5 da chave pública, para fins de autenticação):

> **pgp -kvc**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 15:11 GMT

Key ring: 'c:\pgp\pubring.pgp'
Type Bits/KeyID Date User ID
pub 768/9EC7BF9D 1998/04/27 Aluno1 / Curso de PGP <aluno1@na-

cp.rnp.br>
Key fingerprint = 20 43 D9 D4 56 1A 7E 23 4C 4A C9 D6 7E D3 A2 9E
1 matching key found.

O próximo extrair uma cópia da nossa chave pública do anel público para divulgarmos as pessoas que iremos nos corresponder. Isso é feito com:

> **pgp -kx**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/04/27 14:52 GMT

A user ID is required to select the key you want to extract.
Enter the key's user ID: **aluno1@na-cp.rnp.br**

Extracting from key ring: 'c:\pgp\pubring.pgp', userid "aluno1@na-cp.rnp.br".

Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

Extract the above key into which file? **chave.pgp**

Key extracted to file 'chave.pgp'.

Esse arquivo (chave.pgp) contém nossa chave pública no formato binário. Se tentarmos visualizar o conteúdo desse arquivo teríamos um 'monte de lixo' na tela, ou seja, ele não é passível de visualização. Alguns mailers tem problemas com esse formato. PGP oferece uma opção de codificação, chamada de *armouring* que utiliza somente a faixa de caracteres imprimíveis (7bits e não 8bits como o formato binário). O algoritmo utilizado para o *armour* é o RADIX64. Para tanto, devemos acrescentar a opção *-a* a linha de comando:

> **pgp -kxa**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/04/27 15:03 GMT

A user ID is required to select the key you want to extract.
Enter the key's user ID: **aluno1@na-cp.rnp.br**

Extracting from key ring: 'c:\pgp\pubring.pgp', userid "aluno1@na-cp.rnp.br".

Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

Extract the above key into which file? **chave.asc**

Transport armor file: chave.asc

Key extracted to file 'chave.asc'.

Agora podemos visualizar o conteúdo de chave.asc:

```
Type Bits/KeyID    Date       User ID
pub   768/9EC7BF9D 1998/04/27 Aluno1 / Curso de PGP
<aluno1@na-cp.rnp.br>
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQBtAzVEmKUAAAEDAOiUIlW08J6HN9uTVcY2NhDfJ0Ev4jklrAnQoZHDQjEqkqTs
36OooEz5ZlfFECWyk5g1cbG73Vd9TAJvy3LON9H9g4Gim/teIih951asIBJTQ2F7
e2bYYBttV5Gqnse/nQAFEbQrQWx1bm8xIC8gQ3Vyc28gZGUgUEdQIDxhbHVubzFA
bmEtY3Aucm5wLmJyPokAdQMFEDVEmKZtV5Gqnse/nQEBBpQC/3udZj4LB8eg2s3V
sqFl9zeVZDRxYwdSu3KpWwNFTVOJbzfRTi+B/kxOsqL4GrQpwCkKaYye5vRknCE6
ciZVrFG9akVJjwb/utlEE/gE4JGaZCoVavSCBTVAGEgzwAT7jA==
=t59h
-----END PGP PUBLIC KEY BLOCK-----
```

Só nos resta distribuir o arquivo *chave.** para nossos correspondentes.

3.1.2 Via PGPv5.0 (Unix)

A principal mudança que ocorreu no upgrade da versão 2.6.Xi para as 5.Xi foi uma melhor separação das tarefas, isto é, temos diferentes executáveis para diferentes operações. Tanto, que ao chamarmos pelo nome antigo, recebemos o seguinte aviso:

> **pgp**
> PGP is now invoked from different executables for different operations:
>
> pgpe Encrypt (including Encrypt/Sign)
> pgps Sign
> pgpv Verify/Decrypt
> pgpk Key management
> pgpo PGP 2.6.2 command-line simulator (not yet implemented)
>
> See each application's respective man page or the general PGP

documentation
for more information.

Além disso, o novo PGP implementa mais um algoritmo para geração de chaves públicas o Diffie Hellman/DSS, com maior número de bits e mais segurança para tráfego de mensagens via Internet.
Para gerar seu par de chaves no PGP 5.Xi do Unix, o usuário entra com:

> **pgpk -g**
Choose the type of your public key:
1) DSS/Diffie-Hellman - New algorithm for 5.0 (default)
2) RSA
Choose 1 or 2: **1**

Creating DSS/Diffie-Hellman key.

Pick your public/private keypair key size:
(Sizes are Diffie-Hellman/DSS; Read the user's guide for more information)
1) 768/768 bits- Commercial grade, probably not currently breakable
2) 1024/1024 bits- High commercial grade, secure for many years
3) 2048/1024 bits- "Military" grade, secure for forseeable future(default)
4) 3072/1024 bits- Archival grade, slow, highest security
Choose 1, 2, 3 or 4, or enter desired number of Diffie-Hellman bits
(768 - 4096):

Como na versão antiga, ele pede qual o tamanho da chave. Note que as chaves são maiores e que também é permitido ao usuário definir um tamanho à sua escolha de chave. No caso, basta ele entrar com o tamanho desejado ao invés das opções (1, 2, 3 [default] ou 4) de menu. Note também que embora a opção de algoritmo da chave 1 (DSS/Diffie-Hellman) seja mais seguro ele impede que o usuário se comunique com pessoas que possuam a versão antiga do PGP. Normalmente se gera mais de uma versão da chave (cada usuário pode ter mais de um par de chaves nos seus anéis de chaves), uma RSA e outra DSS.

(768 - 4096): **3**
(Producing a 1024 bit DSS and a 2048 bit Diffie-Hellman key)
You need a user ID for your public key. The desired form for this
user ID is your FULL name, followed by your E-mail address enclosed in
<angle brackets>, if you have an E-mail address. For example:
Joe Smith <user@domain.com>
If you violate this standard, you will lose much of the benefits of
PGP 5.0's keyserver and email integration.

Enter a user ID for your public key:

Agora ele pede para o usuário identificar a chave; o padrão deve ser mantido se o usuário quiser se beneficiar dos servidores de chave e da integração com os clientes

de e-mail.
Servidores de chave são hosts de confiança espalhados pela Internet que guardam as chaves públicas dos usuários. Como se fossem um imenso catálogo telefônico de chaves. Isso auxilia na propagação das nossas chaves públicas. Qualquer pessoa que necessite de uma chave pública que não esteja contida em sua base (anéis) de chaves pode requisitar a esses servidores a respectiva chave.
Num passo seguinte o PGP pede qual o tempo de validade em dias da chave, sendo 0 validade eterna (default).

Enter a user ID for your public key: **Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>**

Enter the validity period of your key in days from 0 - 999
0 is forever (and the default): **0**

O próximo passo é a frase-senha que protegerá suas chave privada.

You need a pass phrase to protect your private key(s).
Your pass phrase can be any sentence or phrase and may have many words, spaces, punctuation, or any other printable characters.
Enter pass phrase:
Enter again, for confirmation:
Enter pass phrase:
We need to generate 438 random bits. This is done by measuring the time intervals between your keystrokes. Please enter some random text on your keyboard until you hear the beep:
0 * -Enough, thank you.
.....******* .....******* .

Keypair created successfully.

Agora ele pede qual a URL do servidor de chaves que se deseja disponibilizar a chave pública recém-gerada. Caso não se queira enviar a chave, basta pressionar a tecla ENTER (<CR>). Como no núcleo o PGP é de uso interno, não colocamos as chaves nos servidores.

If you wish to send this new key to a server, enter the URL of the server, below. If not, enter nothing. **<CR>**

Para visualizar a chave, usamos:

> **pgpk -l**
Type Bits KeyID Created Expires Algorithm Use
sec+ 1024 0x38F7F8BF 1998-04-27 ---------- DSS Sign & Encrypt
sub 2048 0x85FFD67C 1998-04-27 ---------- Diffie-Hellman
uid Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>

1 matching key found

Para visualizá-la com o fingerprint:

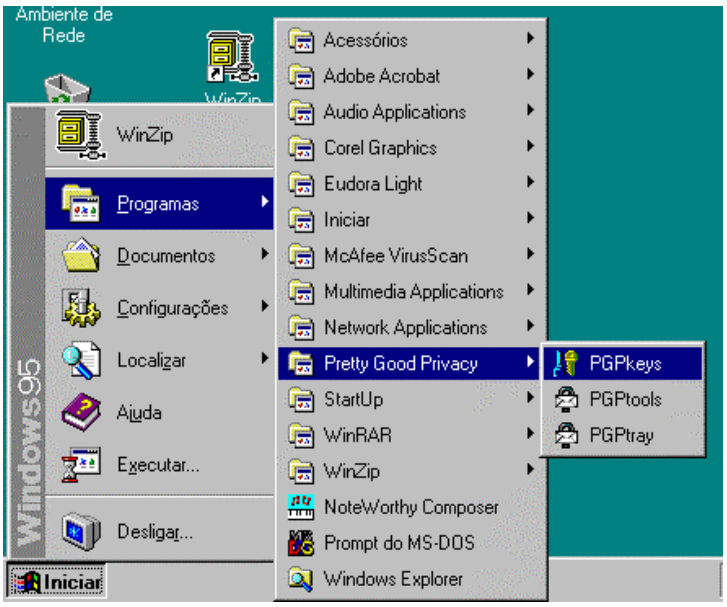
> **pgpk -ll**
Type Bits KeyID Created Expires Algorithm Use
sec+ 1024 0x38F7F8BF 1998-04-27 ---------- DSS Sign & Encrypt
f20 Fingerprint20 = 808C EFDE E533 F6C9 02CF 427B 4F8F 3BDD 38F7 F8BF
sub 2048 0x85FFD67C 1998-04-27 ---------- Diffie-Hellman
f20 Fingerprint20 = 013A 3090 C75E B341 D096 25AA 4A8F E00F 85FF D67C
uid Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>
SIG 0x38F7F8BF 1998-04-27 Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>

Para extrair a chave pública:

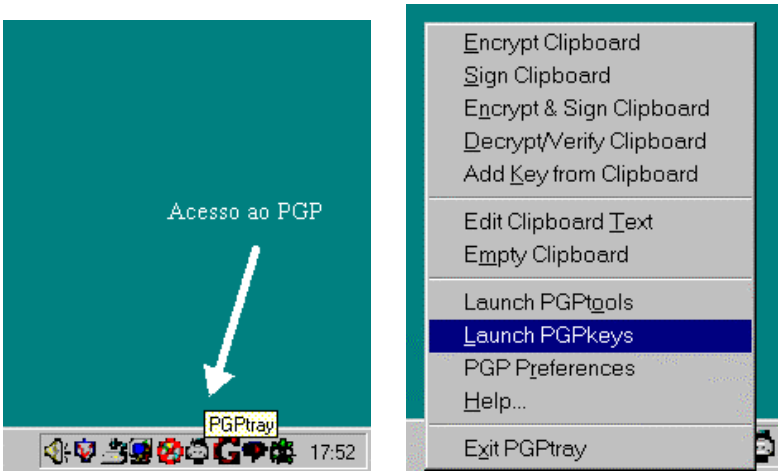
> **pgpk -x aluno1@na-cp.rnp.br -o chave.pgp**
ou
> **pgpk -xa aluno1@na-cp.rnp.br -o chave.asc**

3.1.3 Via PGPv5.5 (Windows)

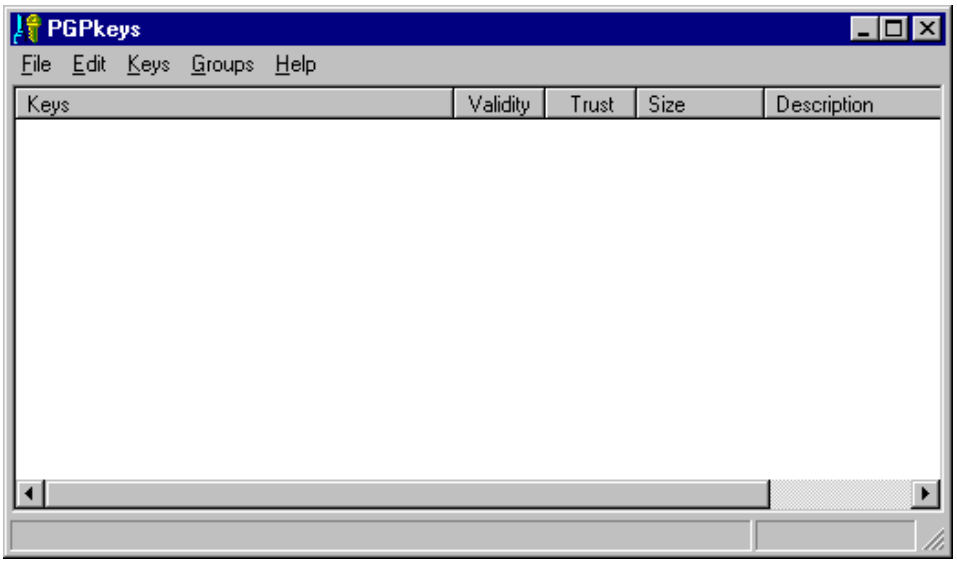
Para criar nosso par de chaves no Windows, necessitamos utilizar o módulo de gerência de chaves. O gerenciador de chaves PGP 5.Xi para Windows pode ser acionado da maneira convencional, via Menu Iniciar->Programas->Pretty Good Privacy->PGPKeys.



Ou via barra de tarefas, ícone PGPTray.



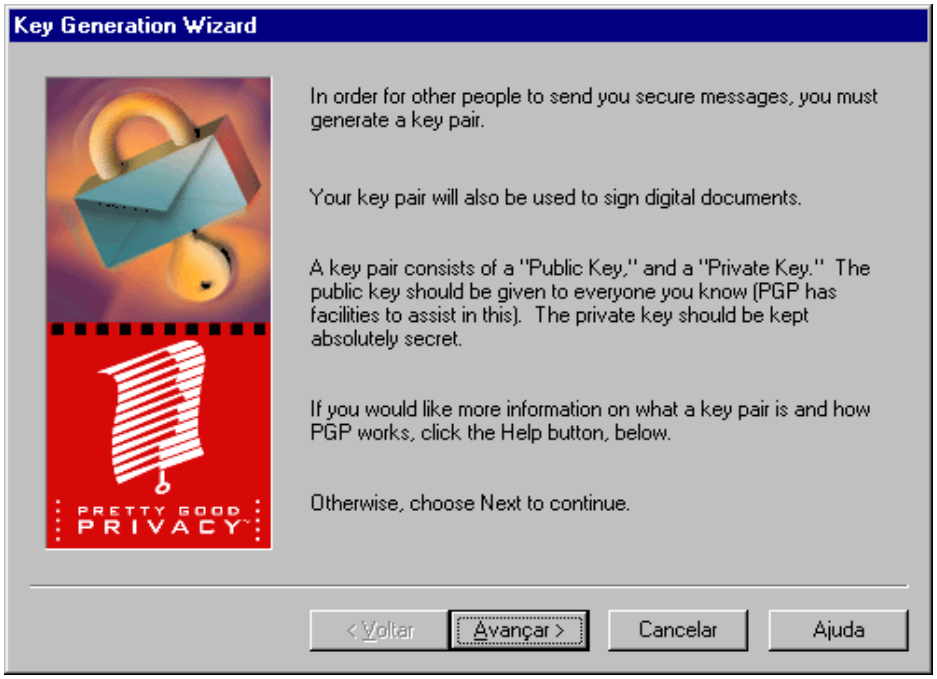
Seremos presenteados com a seguinte interface:



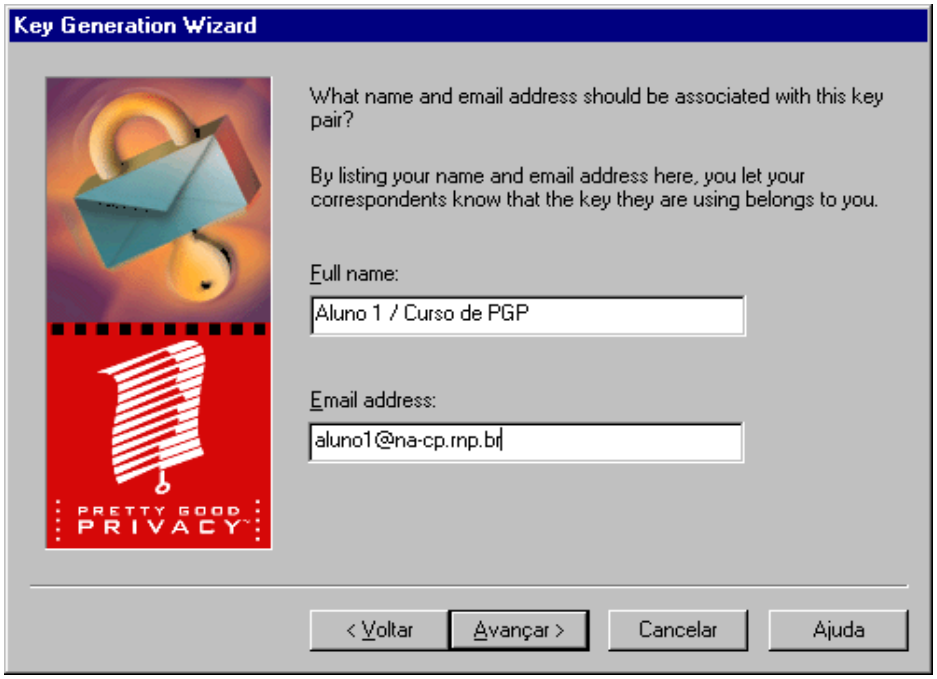
Segue-se breve explicação de cada item do Menu Principal:

- File: Sair do programa
- Edit: Manusear a base de dados, parâmetros de visualização e customização do gerenciador
- Keys: Criação de chaves, extração/importação de chaves públicas, revogar chaves, buscar/enviar de/para servidor, assinar chaves, alterar propriedades da chaves, mudar a seleção das colunas a serem mostradas.
- Groups: Uma das características marcantes do novo PGP. Gerência de Grupos de chaves; ou seja, podemos agrupar nossos recipientes e assim enviar mails destinados a um grupo.
- Help: ajuda, caixa de informações acerta do PGP e atualização para versão comercial.

Para que o PGP gere o par de chaves do usuário, deve-se selecionar o item: *Keys->New Key* (Control-N)
O usuário será então apresentado um "Wizard" (Assistente) para geração de chaves:

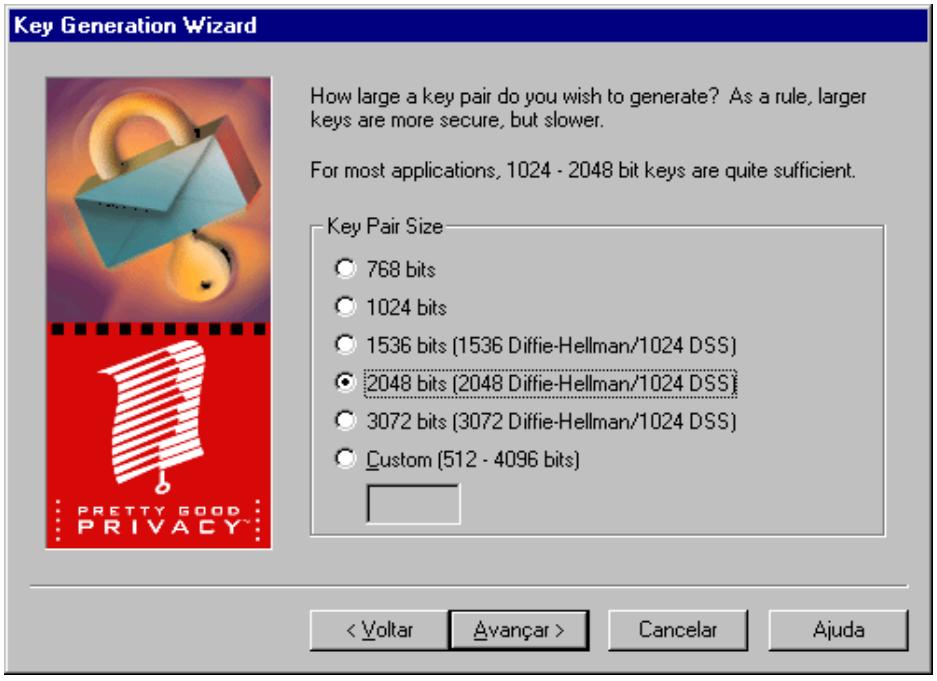


Nada mais simples... Basta seguir as instruções e preencher os campos:

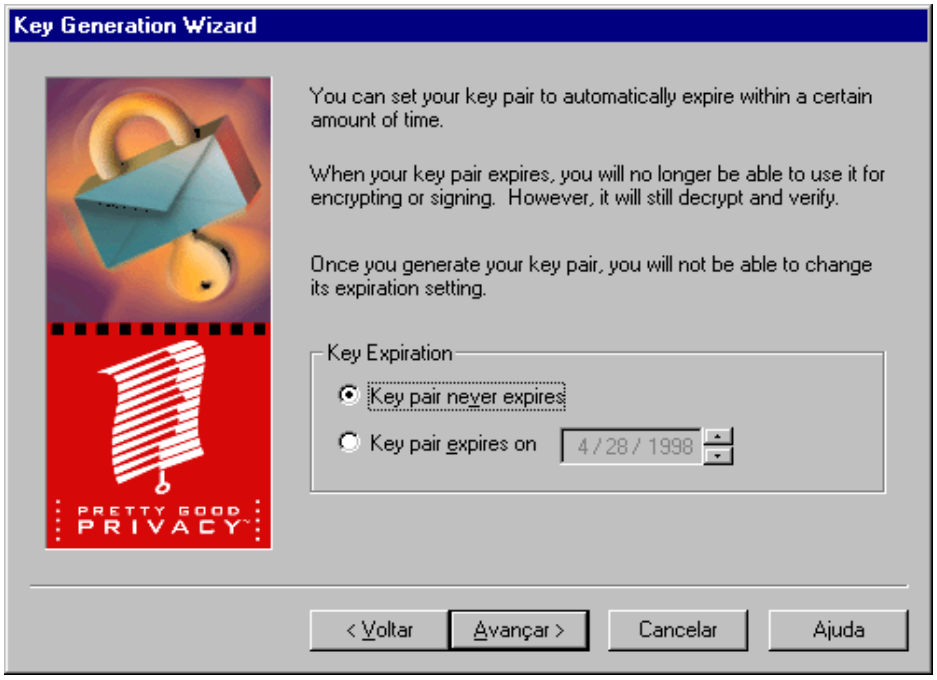


O próximo passo é a escolha do algoritmo para geração de chaves públicas. Lembre-se: Se quiser compatibilidade com o PGP 2.6.Xi crie também uma chave RSA.

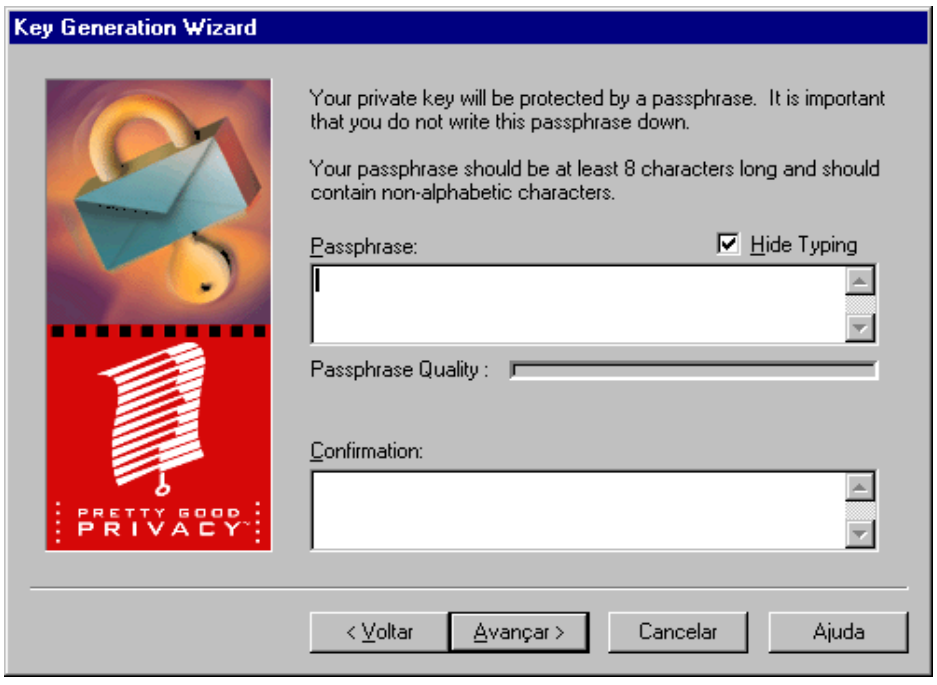
Agora escolha o tamanho das chaves. Se quiser um customizado (particular), escolha o Custom e na caixa de diálogo entre com o valor respeitando a faixa entre parênteses (512-4096 bits).



Em seguida, escolha a data de expiração do seu par de chaves. Mesmo que venha a escolher *nunca*, você pode revogar uma chave a qualquer hora usando o menu: Keys->Revoke.

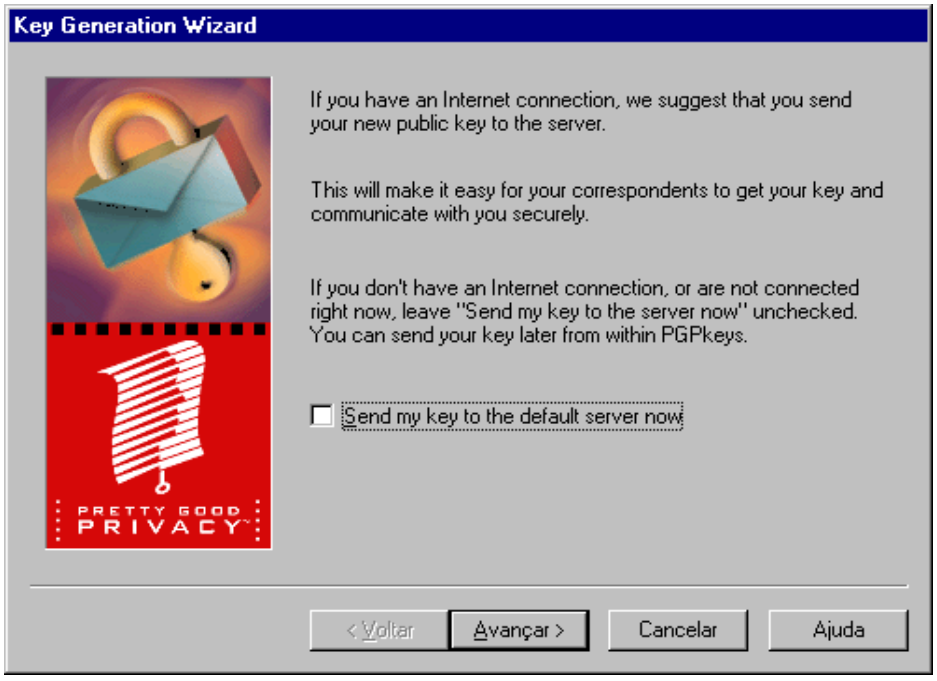


Entre então com a frase-senha que protegerá sua chave secreta. Se tiver dificuldades com a digitação, você pode deschecar o item *Hide Typing* e o texto que digitar sairá em claro. Há um avaliador real-time da qualidade da sua frase-senha (Passphrase quality).

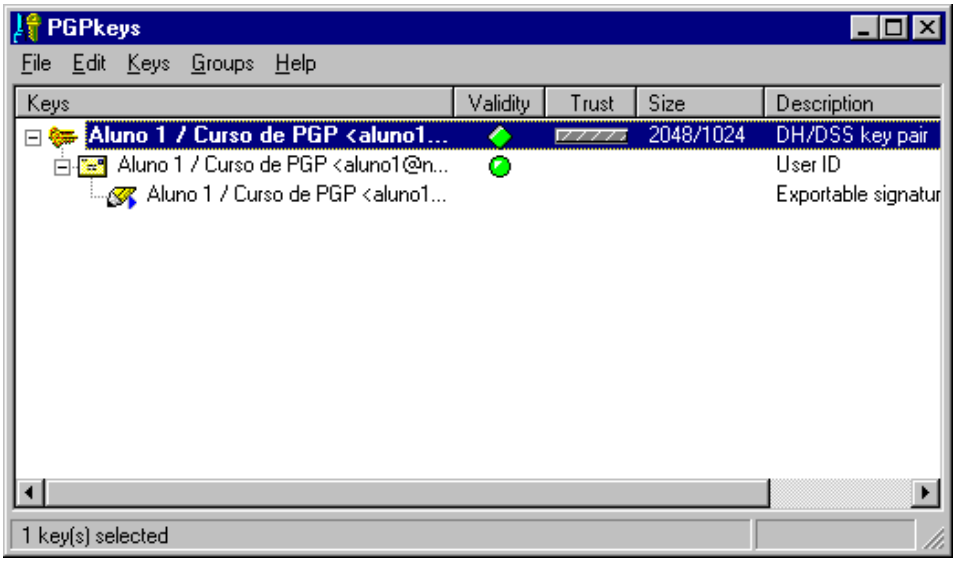


Ele então gera o par de chaves...

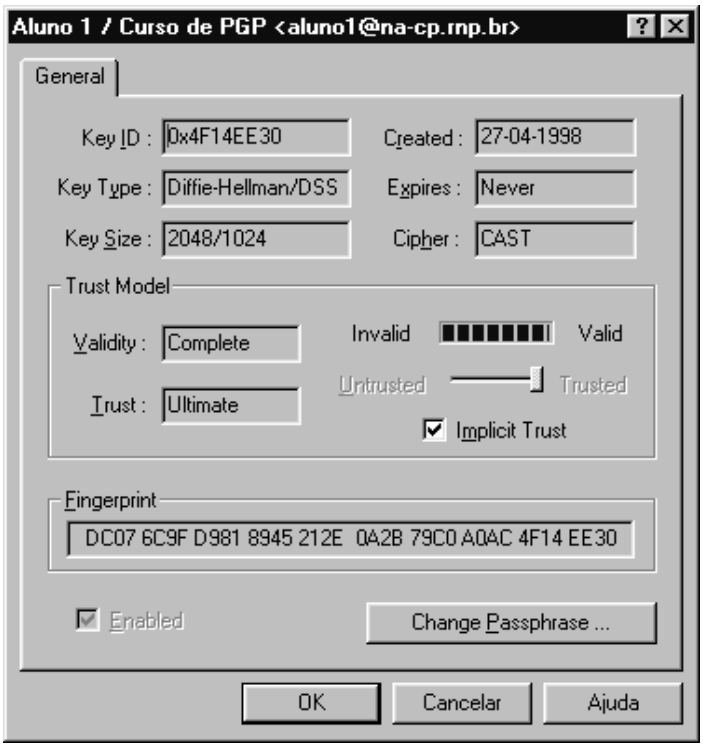
Em seguida, PGP pergunta se o usuário deseja disponibilizar a chave em algum servidor de chaves. Pode-se enviar a qualquer hora, a posteriori via menu: Keys->Send Key To Server.



E voi-lá. Par de chaves geradas.



Para visualizá-la, clique na chave, use o botão direito do mouse e escolha propriedades da chave (Key Properties) ou clique na chave, clique em Keys->Key Properties.



Para extrair sua chave, o usuário clica na chave desejada e seleciona *Keys->Export*. Basta, então, escolher o nome do arquivo da chave e clicar Salvar.

3.2 Manuseando chaves do PGP

3.2.1 Via PGPv2.6.3i (DOS/Unix)

Para adicionar uma chave ao seu Anel de Chaves (=chaveiro, base de chaves, keyring), entre com:

> **pgp -ka chave.asc**
> Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
> (c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
> International version - not for use in the USA. Does not use RSAREF.
> Current time: 1998/05/04 15:23 GMT
>
> Looking for new keys...
> pub 768/D23D1B1D 1997/05/27 Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
>
> Checking signatures...
> pub 768/D23D1B1D 1997/05/27 Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
> sig! D23D1B1D 1997/05/27 Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
>
> Keyfile contains:
> 1 new key(s)

One or more of the new keys are not fully certified.
Do you want to certify any of these keys yourself (y/N)?

No caso, eu estou adicionando a chave pública do usuário Fábio do Núcleo da RNP de Campinas no *KeyRing* de aluno1.
Podemos certificar a chave de Fábio e colocá-lo como nosso introdutor.

Do you want to certify any of these keys yourself (y/N)? **y**

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27
Key fingerprint = 81 9D A7 EA A7 37 78 CD 14 FD 76 C5 DA 17 C7 77
This key/userID association is not certified.
Questionable certification from:
Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>

Do you want to certify this key yourself (y/N)? **y**

Looking for key for user 'Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>':

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27
Key fingerprint = 81 9D A7 EA A7 37 78 CD 14 FD 76 C5 DA 17 C7 77

READ CAREFULLY: Based on your own direct first-hand knowledge, are you absolutely certain that you are prepared to solemnly certify that the above public key actually belongs to the user specified by the above user ID (y/N)? **y**

You need a pass phrase to unlock your RSA secret key.
Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

Enter pass phrase: Pass phrase is good. Just a moment....
Key signature certificate added.

Make a determination in your own mind whether this key actually belongs to the person whom you think it belongs to, based on available evidence. If you think it does, then based on your estimate of that person's integrity and competence in key management, answer the following question:

Would you trust "Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>"
to act as an introducer and certify other people's public keys to you?
(1=I don't know. 2=No. 3=Usually. 4=Yes, always.) ? **4**

Note que poderíamos responder não quando ele pergunta pela certificação, ou bem como definir um nível diferente de confiança acima. O caráter é informativo. E pode ser alterado a posteriori via: *pgp -ke usuario*. E a certificação pode ser alterada via: *pgp -ks usuario.* (ver assinando um chave no seu anel abaixo: *pgp -ks*). Note ainda que se usarmos *pgp -ke nossoid* poderemos mudar nossa frase-senha.

> **pgp -ke aluno1**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 17:50 GMT

Editing userid "aluno1" in key ring: 'c:\PGP\pubring.pgp'.

Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

You need a pass phrase to unlock your RSA secret key.
Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

Enter pass phrase: Pass phrase is good.
Current user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
Do you want to add a new user ID (y/N)? **n**

Do you want to change your pass phrase (y/N)? **y**

Enter pass phrase:

Uma vez adicionada a chave de Fábio, posso encriptar textos para ele decriptar.
Para remover a chave de um usuário, entramos com: *pgp -kr usuario*. Ou se quisermos a eliminação apenas da assinatura: *pgp -krs usuario*.

> **pgp -kr fabio**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 15:36 GMT

Removing from key ring: 'c:\pgp\pubring.pgp', userid "fabio".

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27

Are you sure you want this key removed (y/N)? **y**

Key removed from key ring.

Aliás, falando em assinatura, podemos assinar uma chave, com *pgp -ks usuario*.

> **pgp -ks fabio**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 18:54 GMT

A secret key is required to make a signature.
You specified no user ID to select your secret key,
so the default user ID and key will be the most recently
added key on your secret keyring.

Looking for key for user 'fabio':

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27
Key fingerprint = 81 9D A7 EA A7 37 78 CD 14 FD 76 C5 DA 17 C7 77

READ CAREFULLY: Based on your own direct first-hand knowledge, are
you absolutely certain that you are prepared to solemnly certify that
the above public key actually belongs to the user specified by the
above user ID (y/N)? **y**

You need a pass phrase to unlock your RSA secret key.
Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

Enter pass phrase: Pass phrase is good. Just a moment....
Key signature certificate added.

Make a determination in your own mind whether this key actually
belongs to the person whom you think it belongs to, based on available
evidence. If you think it does, then based on your estimate of
that person's integrity and competence in key management, answer
the following question:

Would you trust "Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>"
to act as an introducer and certify other people's public keys to you?
(1=I don't know. 2=No. 3=Usually. 4=Yes, always.) ? **2**

Para revogar, ou desabilitar uma chave: *pgp -kd usuario*. Usado quando não se confia mais naquela chave, por algum motivo como vazamento de informação, ou qualquer que seja.

> **pgp -kd fabio**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 15:41 GMT

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27

Disable this key (y/N)? **y**

Se entrarmos o mesmo comando de novo, reabilitaremos a chave....

> **pgp -kd fabio**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/04 15:41 GMT

Key for user ID: Fabio Rogerio Hideki Okamura <fabio@na-cp.rnp.br>
768-bit key, key ID D23D1B1D, created 1997/05/27
Key is disabled.

Key is already disabled.
Do you want to enable this key again (y/N)? **y**

3.2.2 Via PGPv5.0 (Unix)

Para adicionar uma chave, usamos: *pgpk -a arquivo1 [arquivo2 arquivo3 ...]*

> **pgpk -a chave.asc**
Adding keys:

Key ring: 'chave.asc'
Type Bits KeyID Created Expires Algorithm Use
pub 768 0x800E8410 1998-01-06 ---------- DSS Sign & Encrypt
sub 769 0x7677B8C9 1998-01-06 ---------- Diffie-Hellman
uid Ranieri Romera <romera@vnet.ibm.com>

1 matching key found

Add these keys to your keyring? [Y/n] **y**

Keys added successfully.

Para remover uma chave: *pgpk -r usuario* ou *pgpk -ru usuario* (esse último remove *usuario* do seu anel público e privado).

> **pgpk -r romera@vnet.ibm.com**

pub 768 0x800E8410 1998-01-06 ---------- DSS Sign & Encrypt
sub 769 0x7677B8C9 1998-01-06 ---------- Diffie-Hellman
uid Ranieri Romera <romera@vnet.ibm.com>

The following key has been selected to be removed:

pub 768 0x800E8410 1998-01-06 ---------- DSS Sign & Encrypt
sub 769 0x7677B8C9 1998-01-06 ---------- Diffie-Hellman
uid Ranieri Romera <romera@vnet.ibm.com>
Removed.

Para alterar nosso grau de confiança em relação a chave de *usuario* de modo a que ele possa falar pela veracidade da nossa chave; isto é, nosso grau de confiança no *usuario* como nosso introdutor: *pgpk -e usuario*

> **pgpk -e romera@vnet.ibm.com**

pub 768 0x800E8410 1998-01-06 ---------- DSS Sign & Encrypt
sub 769 0x7677B8C9 1998-01-06 ---------- Diffie-Hellman
uid Ranieri Romera <romera@vnet.ibm.com>

768 bits, Key ID 0x800E8410, created 1998-01-06
"Ranieri Romera <romera@vnet.ibm.com>"

Would you trust this key owner to act as an introducer and
certify other people's public keys to you?
(1=I don't know. 2=No. 3=Usually. 4=Yes, always)? **2**

Para mudar nossa senha-frase: *pgpk -e nossoid*.

> **pgpk -e aluno1**

sec 1024 0x38F7F8BF 1998-04-27 ---------- DSS Sign & Encrypt

sub 2048 0x85FFD67C 1998-04-27 ---------- Diffie-Hellman
uid Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>

1024 bits, Key ID 0x38F7F8BF, created 1998-04-27
"Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>"

Do you want to unset this key as axiomatic [y/N]? **n**

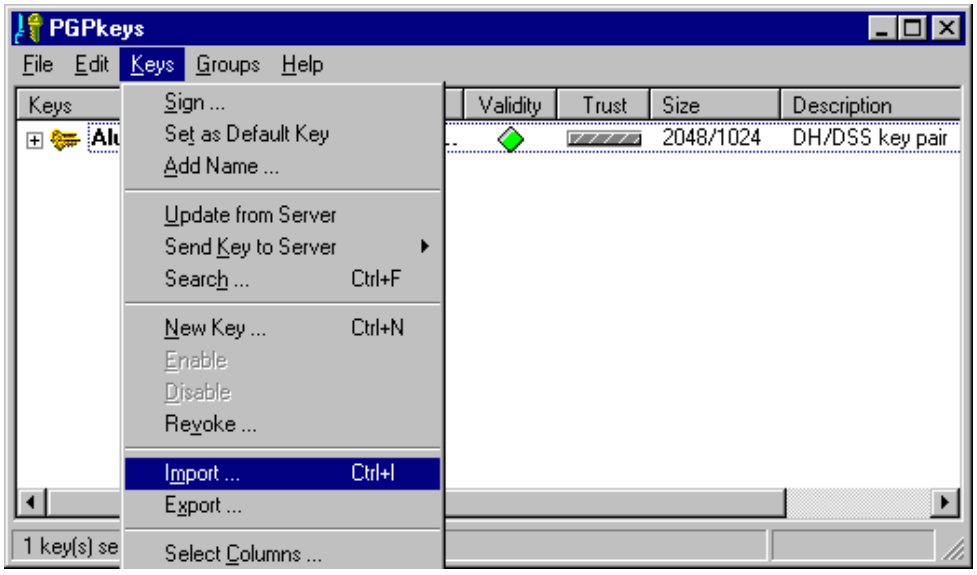
Do you want to add a new user ID [y/N]? **n**

Do you want to change your pass phrase (y/N)? **y**
Need old passphrase. Enter pass phrase:
Need new passphrase. Enter pass phrase:
Enter it a second time. Enter pass phrase:
Changing master key passphrase...

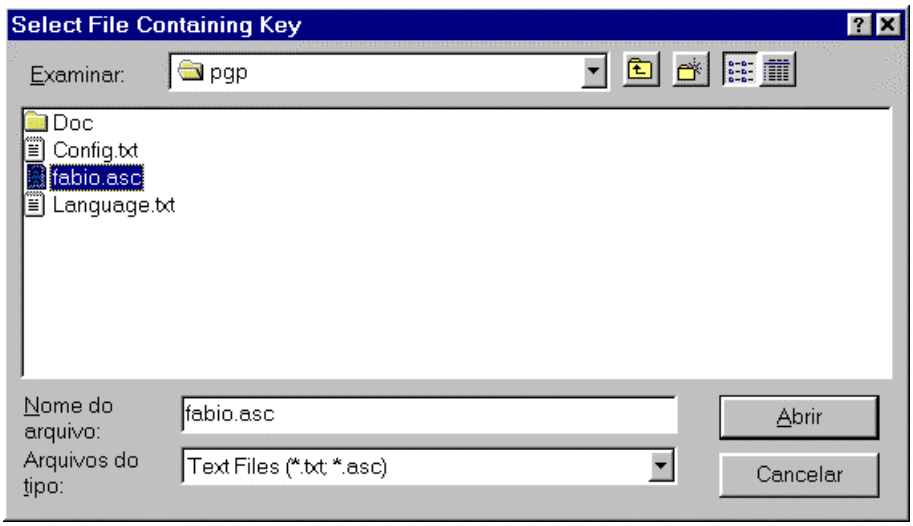
Para assinar uma chave no seu anel: *pgpk -s usuario [-u nossoid]*.
Para remover uma assinatura no seu anel: *pgpk -rs usuario*.
Para desabilitar e reabilitar a chave de *usuario* no seu anel: *pgpk -d usuario*.
Para revogar permanentemente sua própria chave dos anéis públicos e privados: *pgpk --revoke nossoid*.
Para revogar nossa assinatura na chave de algum *usuario*: *pgpk --revokes nossoid*.

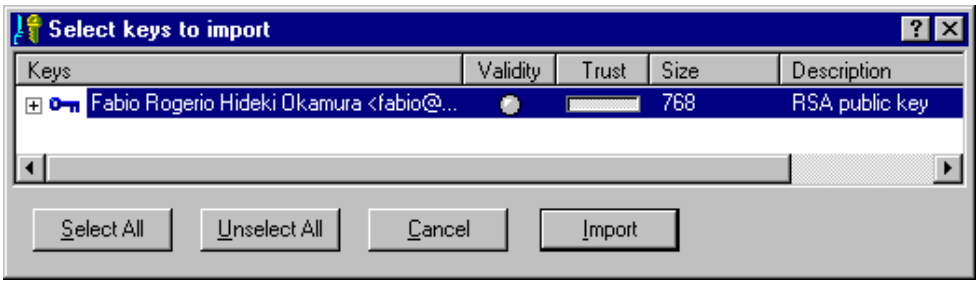
3.2.3 Via PGPv5.5 (Windows)

Para acrescentar uma chave ao nosso pubring: *Keys->Import*.

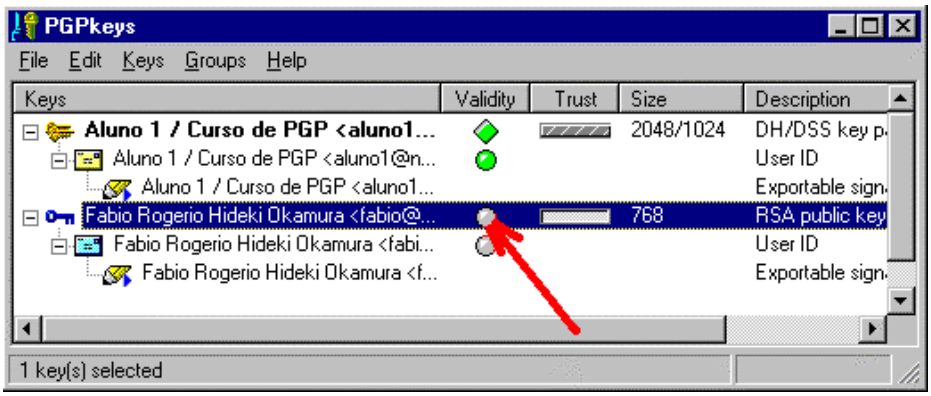


Navegamos pela estrutura de diretórios a fim de ir de encontro ao arquivo *fabio.asc* passado de alguma forma para nós. A fim de exemplificar, pegamos o arquivo contendo a chave do usuário Fábio do Núcleo de Campinas, obtido via finger ou diretamente de sua Home-Page.

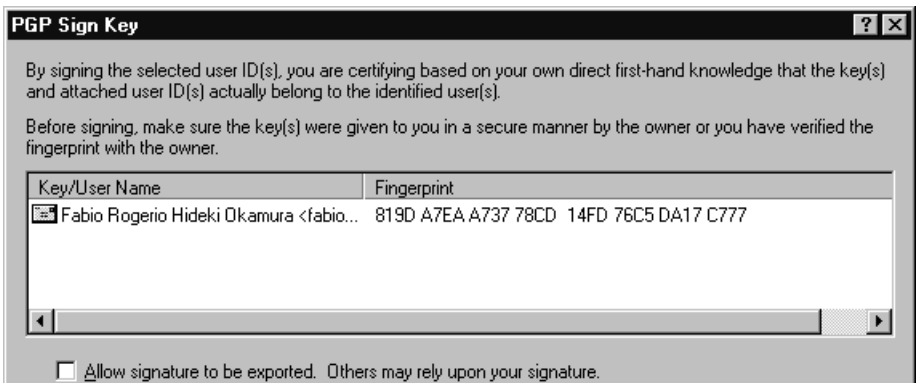




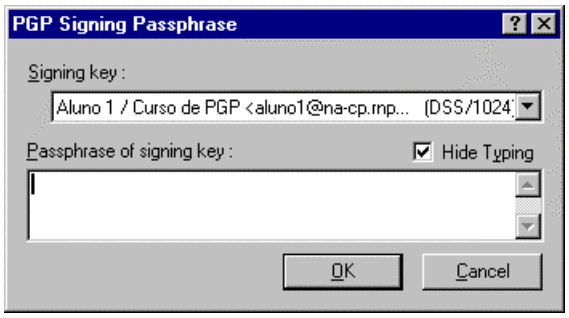
Basta clicar *Import* para efetuar a adição da chave pública de Fábio ao nosso anel. Observe o campo validade e confiança. Para alterarmos, basta fazer como nos PGPs anteriores, ou seja, assinar a chave.
Clique em *Keys->Sign* ou clique na bolinha validade (cinza).



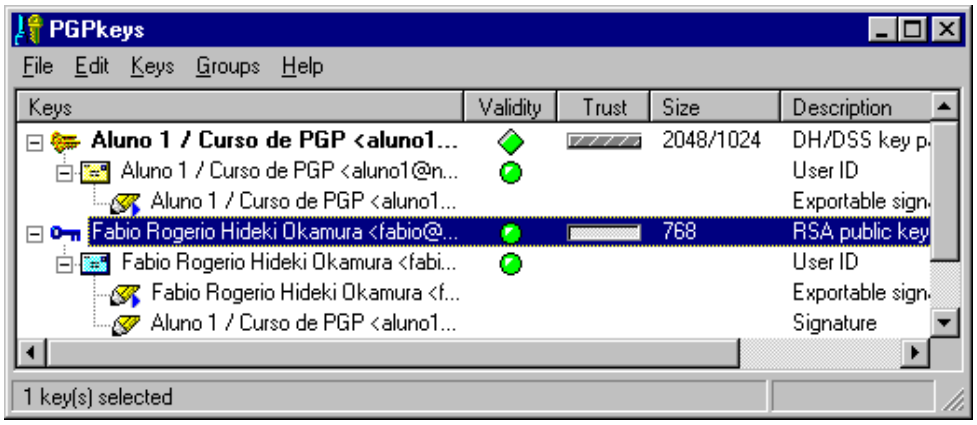
É apresentado o 'fingerprint' do usuário para fins de certificação. (Podemos confirmar com o dono da chave por telefone, por exemplo). Clicando OK, daremos nosso aval a chave.



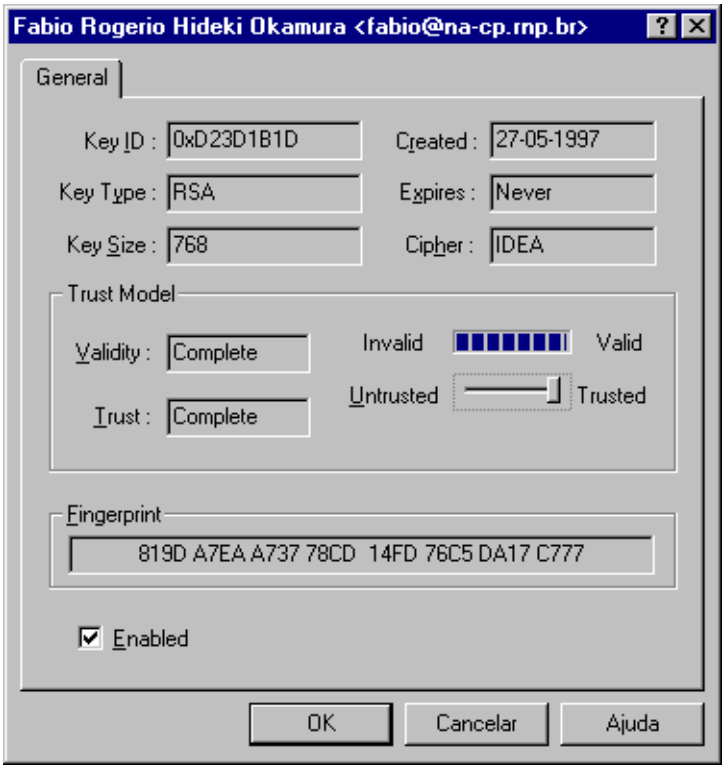
Para isso vamos precisar da nossa senha-frase.



Agora podemos alterar nosso grau de confiabilidade (trust) no usuário para ser nosso introdutor. Basta clicar na chave e então selecionar: *Keys->Key Properties* ou clique com o botão direito e escolha *Key Properties*.



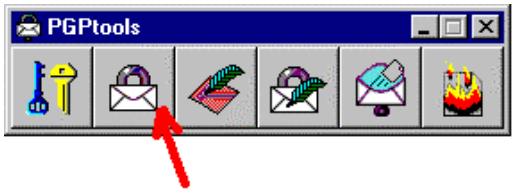
Movemos o slider para alterar o grau de confiança, de Untrusted (equivale ao No=2 do antigo PGP) para Marginal (equivale ao Usually=3) e então Complete(equivale ao Yes,Always=4).



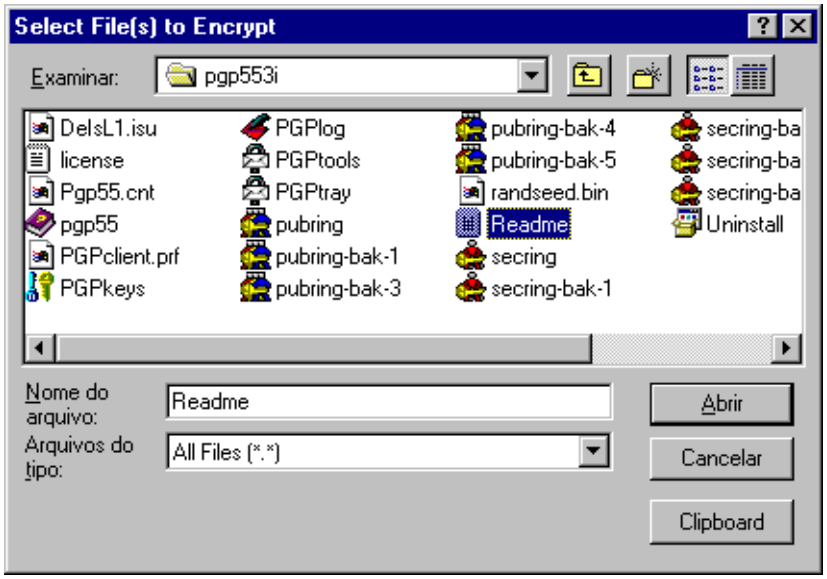
Para remover uma chave, basta selecioná-la e teclar <Del> ou clicá-la com o botão direito e selecionar *Delete*.
Para desabilitar uma chave, basta selecioná-la e usar *Keys->Disable* ou clicá-la com o botão direito e selecionar *Disable*.
Para revogar (inutilizar) nossa chave, basta clicá-la com o botão direito e selecionar *Revoke* ou selecioná-la e usar *Keys->Revoke*.

3.3 Encriptando com o PGP

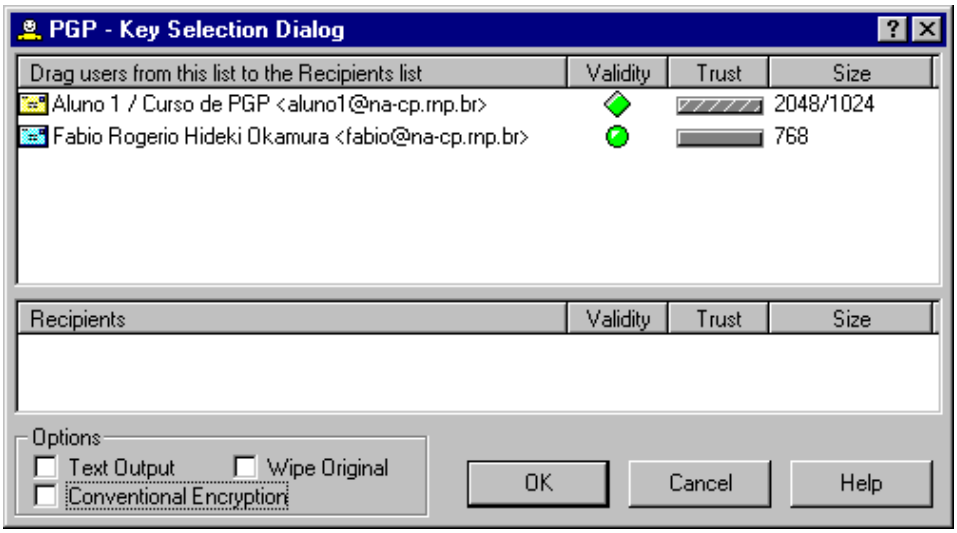
PGP pode ser usado para encriptar arquivos via encriptação convencional - útil para arquivos no nosso 'HardDrive' (arquivamento pessoal e privativo). Nesse caso não é necessário um recipiente. Porém é necessário um segredo, mais propriamente uma frase-chave que será utilizada como base para geração de uma chave de encriptação/decriptação (via MD5).
No PGPv5.5 do Windows, devemos utilizar o PGPTools - disparado a partir do Ícone PGPTray na barra de ferramentas ou a partir do *Menu Iniciar->Programas->Pretty Good Privacy*. Clicando no Segundo Ícone (o do envelope lacrado) do PGPTools.



Selecionamos então o arquivo a ser encriptado, por exemplo o Readme.txt

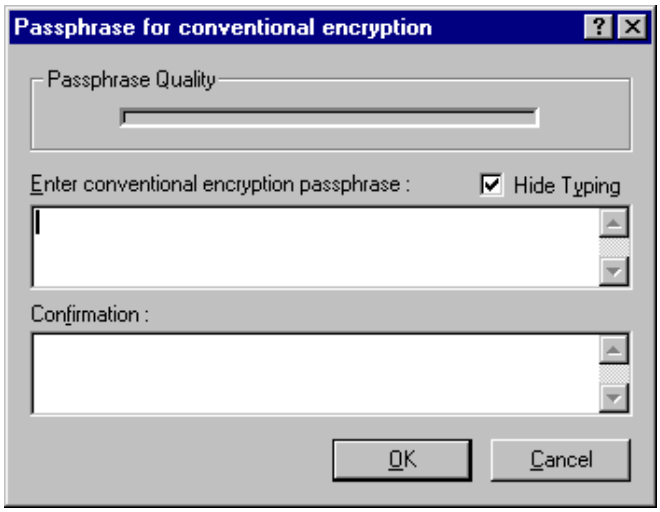


Como queremos encriptar usando o algorítmo da chave única, basta clicar em encriptação convencional (Conventional Encryption). Se quiséssemos encriptar para um determinado usuário via algoritmo de chaves públicas (tipo quando queremos enviar arquivos por e-mail encriptados, uma espécie encriptação de arquivos 'off-line'), ou grupo de usuários, ou usuários, tudo que devemos fazer é arrastá-los da janelinha superior (users), para a janelinha inferior (recipients), ao invés de selecionar *Conventional Encryption*.



Vale notar ainda os 'checkbox'es *Text Output* e *Wipe Original*. Por 'default' (ação padrão), o PGP gera uma saída binária (não pode ser vista por editores ou clientes de e-mail). Para gerar uma saída, digamos, mais textual, aciona-se a opção *Text Output*. Equivale a opção *-a* dos PGPs anteriores. A opção *Wipe Original* elimina o arquivo depois de encriptado. Equivale a opção *-w* do PGP 2.6.3i. Observe que não é um mero delir de arquivos. Cabe aqui uma analogia: *delete* está para uma cesta de lixo de papéis, assim como *wipe* está para uma trituradora de papéis. O arquivo é sujo com códigos sem sentido e depois removido (para que bisbilhoteiros escovadores de bits não encontrem resquícios do arquivo no seu 'hard-drive'). Podemos remover o original a qualquer hora via ícone da carta queimando (o

último ícone) do menu do PGPTools.
Ao clicarmos *Conventional Encryption* e *OK*; seremos indagados da frase-senha **daquele** arquivo, somente. Não é boa política de segurança utilizar ou a frase-senha que protege seus anéis ou encriptar vários arquivos com a mesma frase-senha.



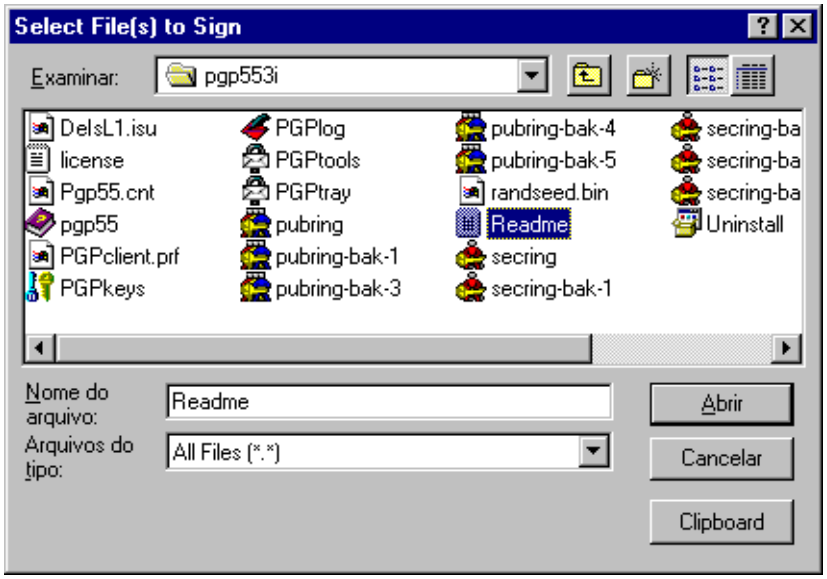
Lembre-se, a saída será: *arquivo.pgp* a não ser que tenhamos optado por uma saída textual, quando então a saída será: *arquivo.asc*.
A mesma coisa pode ser feita nos PGPs antigos via linha de comando: *pgp -c arquivo* (PGPv2.6.3i) ou *pgpe -c arquivo* (PGPv5.0i). Para saída textual, acresentamos um *-a*: *pgp -ca arquivo* (PGPv2.6.3i) ou *pgpe -ca arquivo* (PGPv5.0i). Para eliminar (wipe) o arquivo: pgp -cw arquivo (PGPv2.6.3i, somente).

Se quiséssemos encriptar para algum destinatário (via chaves públicas), devemos utilizar: *pgp -e arquivo usuariodestino1 [usuariodestino2 ...]* ou com saída textual: *pgp -ea arquivo usuariodestino1 [usuario destino2 ...]* (PGPv2.6.3i). Para o PGPv5.5: *pgpe -r usuariodestino1 [-r usuariodestino2 ...] arquivo* ou para uma saída textual: *pgpe -r usuariodestino1 [-r usuariodestino2 ...] -a arquivo*. Podemos fornecer nomes alternativos de saída via *-o nomealternativo*. Então, basta-nos enviar o arquivo textual de saída para nossos recipientes. Porém o PGP oferece recursos de integração aos clientes de e-mail que facilitam todo o processo, deixando o processo quase invisível ao usuário.

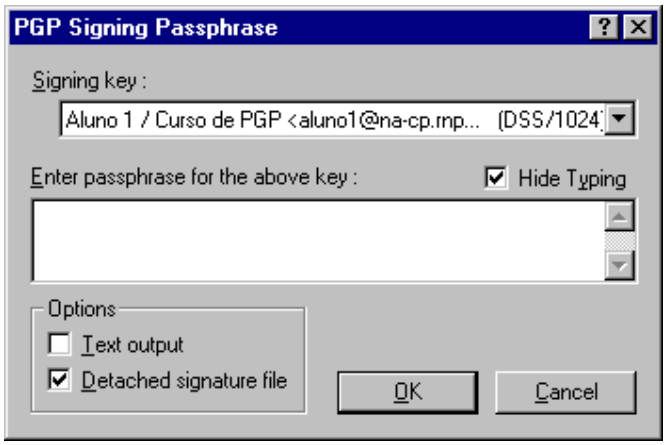
3.4 Assinando com o PGP

Como dito anteriormente, o PGP pode ser usado para fins de autenticação (verificação da veracidade do remetente) via assinaturas digitais. Para assinarmos com o PGPv5.5 do Windows, usamos o ícone da pena assinando uma carta.

Escolhemos o arquivo, por exemplo ReadMe.txt.



E então o sistema pede nossa senha-frase para recuperar nossa assinatura do nosso chaveiro-secreto (anel-secreto, base de chaves secreta, secret keyring).



Note os dois 'checkbox'es. Text Output é nosso velho conhecido. Ele gera uma saída textual ao invés da binária. A novidade é a segunda caixa. Se ativa ela simplesmente gera o bloco da assinatura digital para ser a posteriori anexada pelo usuário ao final da mensagem, caso contrário ela gera um bloco único com mensagem e assinatura.
Suponha a mensagem:

> **type mensa.txt**
oi
esta eh uma mensagem.
foi feita apenas para testar
o pgp.

Com *Text Output* e *Detached signature file* ativos:

> **type mensa.txt.sig**
-----BEGIN PGP SIGNATURE-----

Version: PGPfreeware 5.5.3i for non-commercial use
<http://www.pgpi.com>

iQA/AwUANU8S5HnAoKxPFO4wEQKSMwCfYxeH23daRdovGqGvqZY
zjUeuWX8AoMMx
e6WvIC5XKJwmRmRACL4zMvzb
=p307
-----END PGP SIGNATURE-----

Somente *Text Output* ativo:

> **type mensa.txt.asc**
-----BEGIN PGP MESSAGE-----
Version: PGPfreeware 5.5.3i for non-commercial use
<http://www.pgpi.com>

owHrZLBnZmUw9Rd2rjywYI2/yDsDQSamGQzzPVg/7LkoVFUQOvGS+f
qy9INNlnJN
DHOFDndHSlzcyb+o2rfTfdnxw7URrOxrApI4c1PzihP1SipKGIAgP5OXK
7W4JFEh
NUOhNDdRASyZnpqrx8uVlp+pkJaaCZRLLEjNSyxWKEgsSlQoAaku4uX
KVyhILwCq
AgA=
=sZ9D
-----END PGP MESSAGE-----

Obviamente não há sentido em mostrar *Text Output* inativo, pois ele não é 'passível de impressão' pois é binário.
Se quiséssemos assinar e encriptar simultâneamente, optaríamos pelo ícone da carta lacrada sendo assinada por uma pena. O resultado seria um bloco único contendo texto mais assinatura numa PGP message (Vide tópico como funciona o PGP, capítulo 2, item 3).

Para assinarmos com os PGPs anteriores:
Via PGPv2.6.3: *pgp -s arquivo* ou para saída textual: *pgp -sa arquivo* (esses geram um bloco único de arquivo+assinatura); para obter dois blocos: mensagem seguida da assinatura da mensagem, use: *pgp -st arquivo* ou para versão textual: *pgp -sta arquivo*.
Seja a mensagem anterior. Processada por *pgp -sa mensa.txt*:

```
-----BEGIN PGP MESSAGE-----
Version: 2.6.3ia

owHrZChlZmUw9Ze4kRs+cdW84/vnMjIm7GJmOKVk/8yoNrsvS+/AvA6Tjs8nXi1u
/iB17MyirM7m+wsnLvmo3Sz5clPbzqzo8GW7PWdN7Dp+y2+hd+65FvXkjPNfX4Ue
lF91uz4+ln9dpPmiBudy9zPaoVPKReqablkmOCie+aK+JiCJMzc1rzhRr6SihAEI
8jN5uVKLSxIVUjMUSnMTFcCS6am5erxcafmZCmmpmUC5xILUvMRihYLEokSFEpDq
Il6ufIWC9AKgKgA=
=+HBt
-----END PGP MESSAGE-----
```

A mesma mensagem processada por *pgp -sta mensa.txt*:

```
-----BEGIN PGP SIGNED MESSAGE-----

oi
esta eh uma mensagem.
foi feita apenas para testar
o pgp.

-----BEGIN PGP SIGNATURE-----
Version: 2.6.3ia
Charset: cp850

iQB1AwUBNU8ZEG1Xkaqex7+dAQGn4wMA3sQF0+jIp68Xrw13uQg777yCFn5OvDKg
AW3fCFp8hnRlSHhfRtN0Drr8rjL4NnHwIktF5oha/ph/Bt4Z/3SI94l01Qiccn4Z
nMbhTfCj9rFcPfjTCvPfxq6QPt83Zrua
=HhsM
-----END PGP SIGNATURE-----
```

Para o PGPv5.0i usamos o módulo de assinaturas: pgps
Para assinar: *pgps arquivo* (gera uma saída .pgp binária) ou *pgps -a arquivo* (gera saída textual .asc).
Para gerar uma saída com os blocos de textos e assinaturas separadas: *pgps -t arquivo* (saída binária) ou *pgps -ta arquivo* (saída textual).
Exemplo da execução do *pgps -a mensa.txt*:

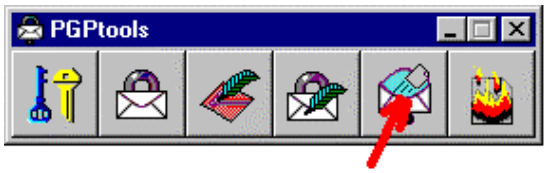
```
-----BEGIN PGP MESSAGE-----
Version: PGP for Personal Privacy 5.0
MessageID: OfPSsJBmK0E9gYNx+gqoSMZ+hcpGHDn0

owHrZLBnZmUw9Zfa4N9vfdfi+4/9gkx22xnmcD4orRFY4fSiTmVnmma129wD90Rf
MCw44sBzYrmF4tULZlffCap9VZ11tKpqjU8SZ25qXnGiXklFCQefR0F+JldqcUmi
QmqGQmluogJYLj01V48rLT9TIS01EyiVWJCal1isUJBYlKhQAlJcxJWvUJBeoMcF
AA==
=P6q+
-----END PGP MESSAGE-----
```

Exemplo da saída da execução do *pgps -ta mensa.txt*:

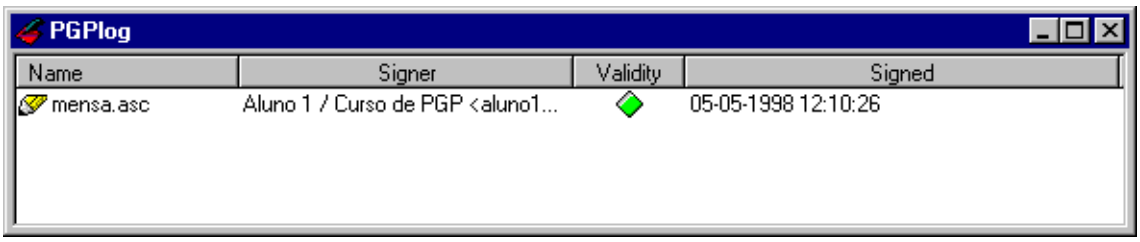
```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

oi
esta eh uma mensagem.
foi feita apenas para testar
o pgp.

-----BEGIN PGP SIGNATURE-----
Version: PGP for Personal Privacy 5.0
Charset: noconv

iQA/AwUBNU8bR0+PO9049/i/EQJT2gCfSzxlpQh299W0fteKzrDj2NPZhjwAoPb0
Bk5R6kBvrdHCYaO6C/Vn8ltV
=4zy0
-----END PGP SIGNATURE-----
```

3.5 Decriptando e Verificando Assinaturas

Para decriptar/verificar a assinatura de um arquivo, no PGPv5.5 do Windows, tudo que temos que fazer é lançar ('launch'=executar, disparar) o PGPTools e clicar no ícone da mensagem saindo do envelope.



Basta navegar e escolher qual o arquivo assinado ou arquivo decriptado ou assinado/decriptado desejamos decriptar e/ou verificar a autenticidade. Supondo que assinamos o arquivo *mensagem.txt* e o arquivo resultante for *mensa.asc*, se abrirmos com o *Decrypt/Verify* obteremos uma janela *PGPlog* que nos informa da veracidade da assinatura. Além disso, essa janela também nos informa se a chave está habilitada ou desabilitada, se está revogada ou não, etc.



Para os PGPs anteriores, o comando é:
Para PGPv2.6.3i, *pgp -d* (decripta, mantêm os dados PGP) *pgp -p* (checa assinatura).
Supondo que assinamos um arquivo *mensa.txt* e precisamos verificar se fomos nós mesmos os proprietários do arquivo:

> **pgp -p mensa.asc**
C:\TEMP\pgpcourse>\pgp\pgp -p mensa.asc
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/05 15:27 GMT
.
File has signature. Public key is required to check signature.
.
Good signature from user "Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>".
Signature made 1998/05/05 15:26 GMT using 768-bit key, key ID 9EC7BF9D

Plaintext filename: mensa.txt

Agora vamos decriptar um arquivo *mensa.txt.asc* que encriptamos para nós mesmos (lembre que precisamos da chave pública do destinatário no caso de envio para outra pessoa):

> **pgp -d mensa.txt.asc**
Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
(c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
International version - not for use in the USA. Does not use RSAREF.
Current time: 1998/05/05 15:32 GMT

File is encrypted. Secret key is required to read it.
Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
768-bit key, key ID 9EC7BF9D, created 1998/04/27

You need a pass phrase to unlock your RSA secret key.
Enter pass phrase: Pass phrase is good. Just a moment.....
Plaintext filename: mensa.txt

Podemos decriptar 'for our eyes only' com a opção *-m*.

> **pgp -dm mensa.txt.asc**
> Pretty Good Privacy(tm) 2.6.3ia - Public-key encryption for the masses.
> (c) 1990-96 Philip Zimmermann, Phil's Pretty Good Software. 1996-03-04
> International version - not for use in the USA. Does not use RSAREF.
> Current time: 1998/05/05 15:48 GMT
>
> File is encrypted. Secret key is required to read it.
> Key for user ID: Aluno1 / Curso de PGP <aluno1@na-cp.rnp.br>
> 768-bit key, key ID 9EC7BF9D, created 1998/04/27
>
> You need a pass phrase to unlock your RSA secret key.
> Enter pass phrase: Pass phrase is good. Just a moment.....
>
> Plaintext message follows...
> ---
>
> oi
> esta eh uma mensagem.
> foi feita apenas para testar
> o pgp.
>
> Save this file permanently (y/N)? n

Para o PGPv5.0 do Unix, usamos para decriptar/verificar assinaturas: *pgpv arquivo*, se não quisermos que o PGP processe assinaturas contidas no arquivo (normalmente PGPv5.0 adicionará qualquer chave que encontrar no arquivo de entrada ao seu KeyRing): *pgpv -K arquivo.*
Vamos verificar nossa assinatura num arquivo assinados por nós mesmos:

> **pgpv mensa.txt.asc**
> Opening file "mensa.txt" type text.
> Good signature made 1998-05-05 13:59 GMT by key:
> 1024 bits, Key ID 38F7F8BF, Created 1998-04-27
> "Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>"

Se quisermos visualizar com um paginador a mensagem assinada: *pgpv -m arquivo*.

> **pgpv -m mensa.txt.asc**
> Opening file "_CONSOLE" type text.
>
> Plaintext message follows...
> ---

oi
esta eh uma mensagem.
foi feita apenas para testar
o pgp.

Done...hit any key

Good signature made 1998-05-05 13:59 GMT by key:
1024 bits, Key ID 38F7F8BF, Created 1998-04-27
"Aluno 1 / Curso de PGP <aluno1@na-cp.rnp.br>"

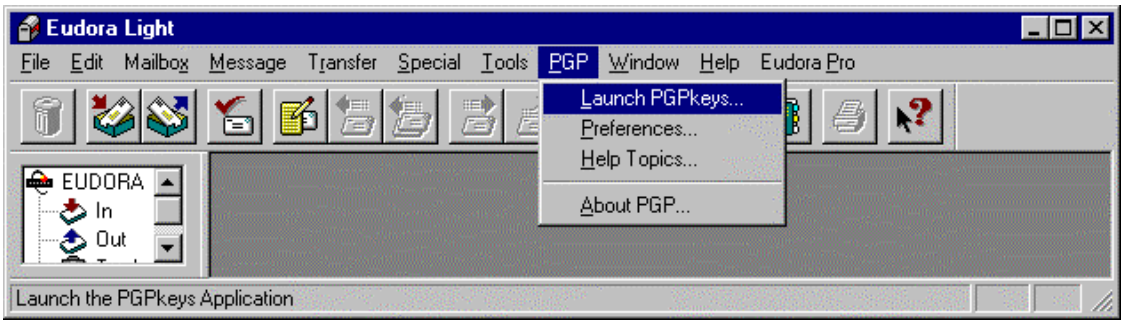
Decriptando uma mensagem encriptados por nós mesmos a nós mesmos (narcisista, não?):

> **pgpv mensa.txt.pgp**
Message is encrypted.
Need a pass phrase to decrypt private key:
2048 bits, Key ID 85FFD67C, Created 1998-04-27
Enter pass phrase:
Pass phrase is good.
Opening file "mensa.txt"

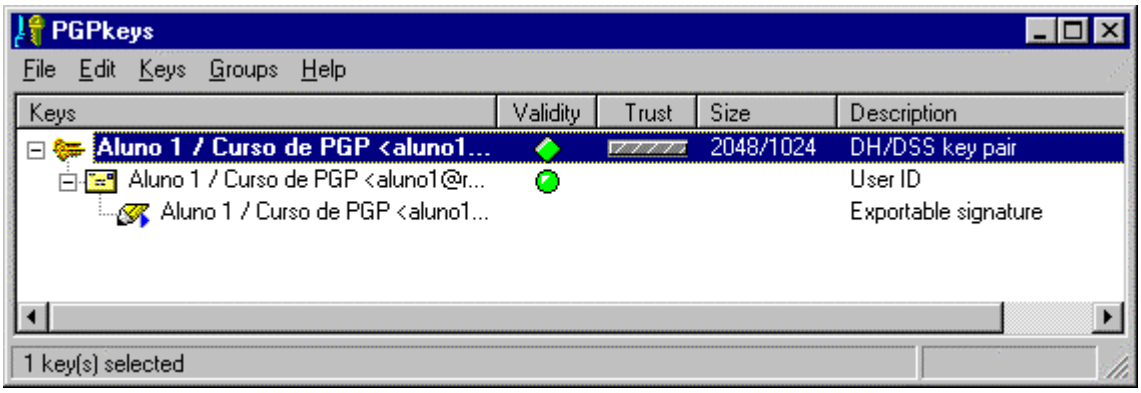
4. PGP integrado aos clientes de e-mail

4.1 Integrado ao Eudora

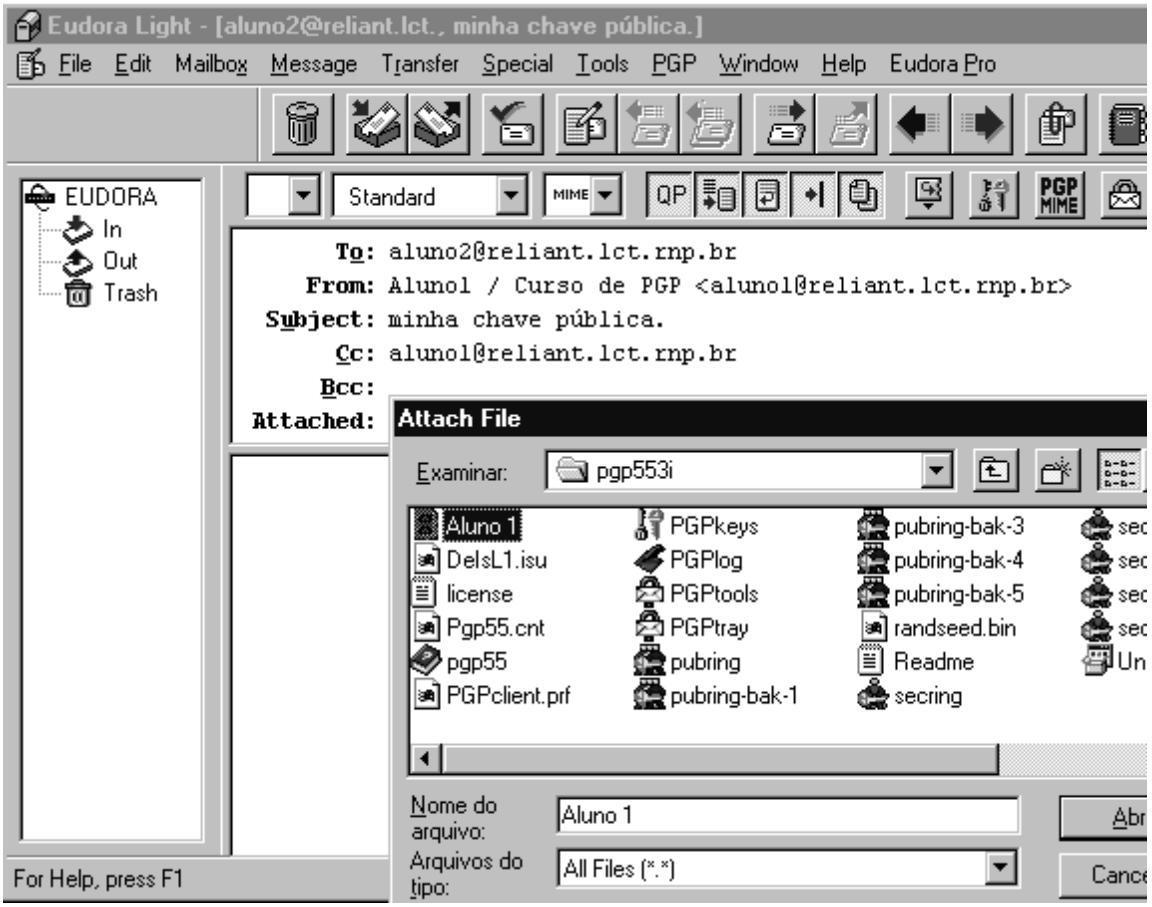
A interface de comunicação do PGP com o Eudora é feita através de Plug-Ins. Ao abrirmos o Eudora, já notamos algumas inclusões feitas pelo plug-in.
A partir do menu PGP nós podemos criar nossas chaves ou adicionar algumas de última hora (Launch PGPKeys). Podemos, também, alterar as configurações do PGP (Preferences); ou consultar a ajuda on-line (Help Topics). E, por que não, um pouco de 'marketing' pessoal (About PGP).



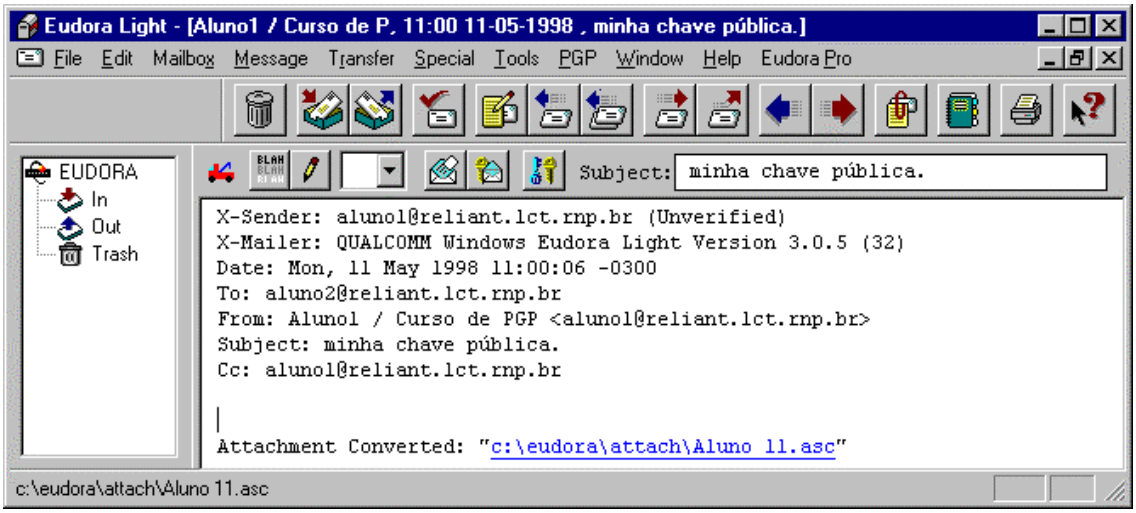
Para fins de demonstração do uso do plug-in, vamos criar um par de chaves para nós (*aluno1@reliant.lct.rnp.br*, no exemplo), para isso, devemos executar o PGPKeys (primeiro item do menu PGP). Consulte o Capítulo 3, item 1, subitem 3.



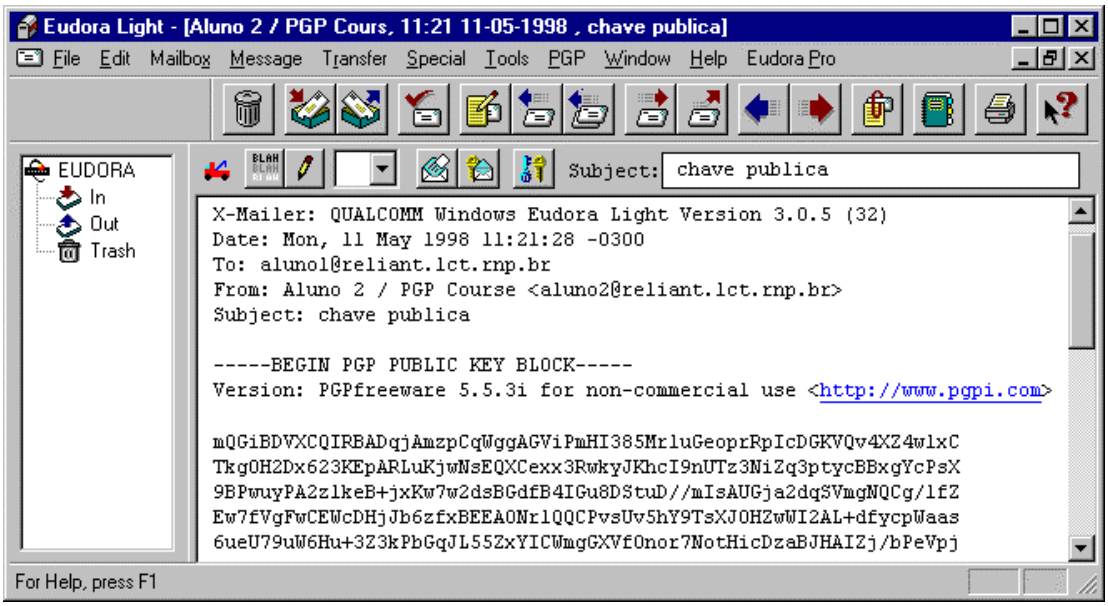
Agora enviamos uma cópia de nossa chave pública para nosso correspondente (*aluno2@reliant.lct.rnp.br*, no exemplo). E recebemos a chave pública deste para anexar-mos em nosso *KeyRing*.



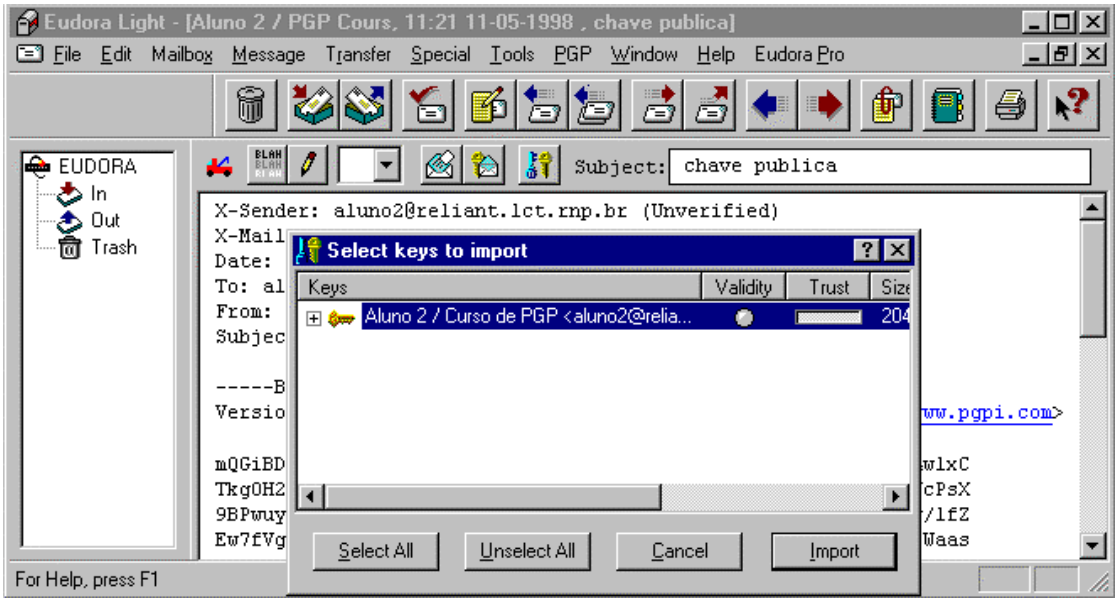
No recebimento de uma chave 'attach'ada, tudo que temos que fazer para inseri-la em nosso *KeyRing* é um duplo-clique no "*Attachment Converted*". Vide Curso de Eudora para detalhes sobre *attachments*.



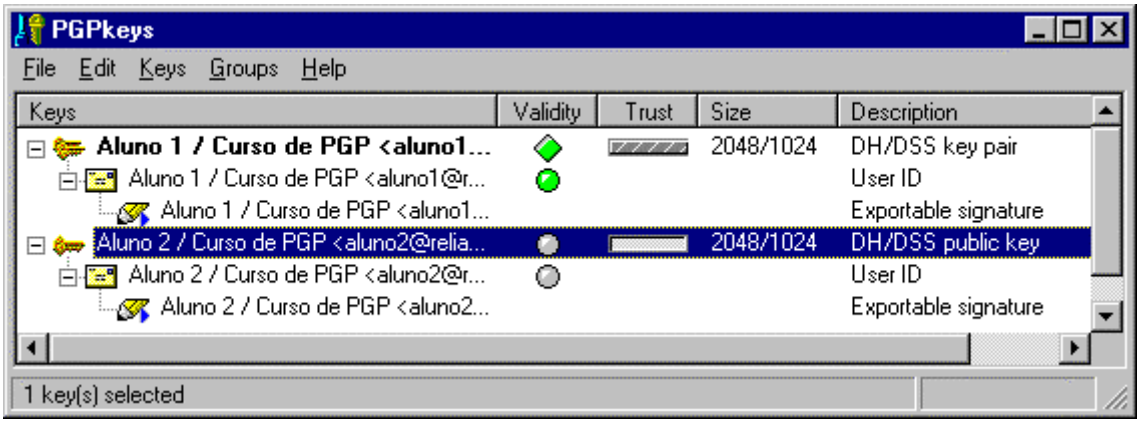
Ou podemos navegar no menu *Keys->Import* do PGPKeys até encontrar o caminho referido ("c:\eudora\attach\Aluno_11.asc", no exemplo acima). Vide Capítulo 3, Item 2, Subitem 3.
Agora, vamos anexar a chave pública de Aluno 2 (*aluno2@reliant.lct.rnp.br*).
Note que a chave nos foi enviada 'plain-text'. Para anexarmos uma chave que tenha sido recebida dessa forma, utilizamos o ícone:  (Extract PGP key from e-mail message).



Quando clicamos o referido ícone:



Já somos presenteados com a caixa de diálogo de Importação de chaves.
Clicamos *Import*, e abrimos o PGPKeys para verificar nosso *KeyRing* (podemos chamá-lo rapidamente através do ícone - Launch PGPKeys Application).



Vamos dar uma breve explanação dos novos ícones introduzidos na interface do Eudora pelo plug-in do PGP.
Na janela de Composição:

Como visto anteriormente no capítulo, o ícone dispara o Gerenciador de Chaves do PGP.

O ícone instrui o PGP a formatar a mensagem 'PGP'ficada em conformidade ao padrão MIME (Multipurpose Internet Mail Extensions: a grosso modo, trata-se de um conjunto de especificações para troca de mensagems entre as várias codificações de caracteres, bem como arquivos multimedia entre diferentes máquinas e diferentes sistemas operacionais que se utilizem do sistema de e-mail da Internet - Vide Referências (Capítulo 7) para maiores esclarecimentos com relação a esse padrão).

 instrui o PGP a encriptar a mensagem antes de enviá-la.

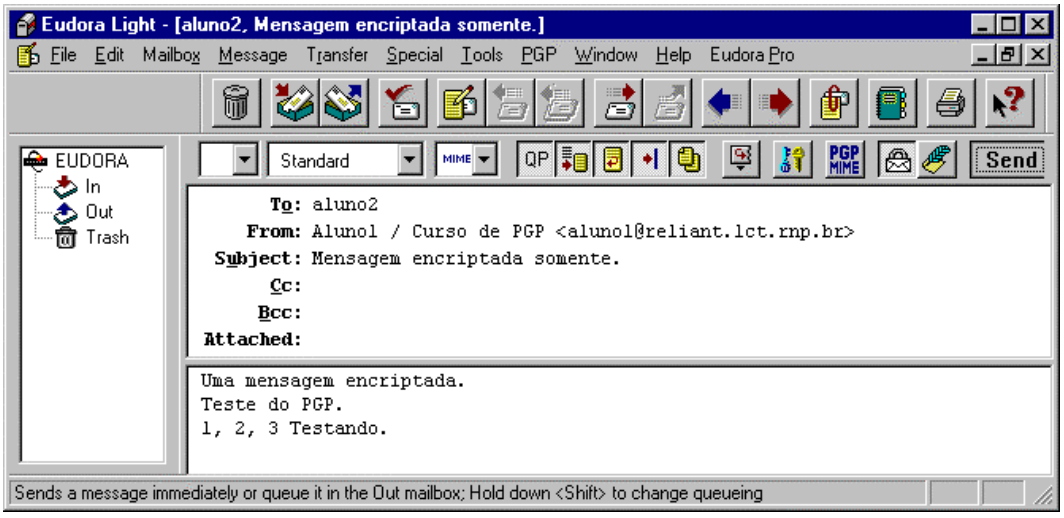
E o ícone  instrui o PGP a assinar a mensagem antes de enviá-la.
Observe que, diferentemente do Pine, é possível qualquer combinação entre encriptação e assinatura, isto é: podemos enviar assinado somente, encriptado somente, ou encriptado e assinado. Basta clicarmos os ícones de acordo com a combinação que desejarmos antes de clicar o botão *Send*.
Na janela de Recepção, temos:

 idem à janela de composição; lança o PGPKeys.

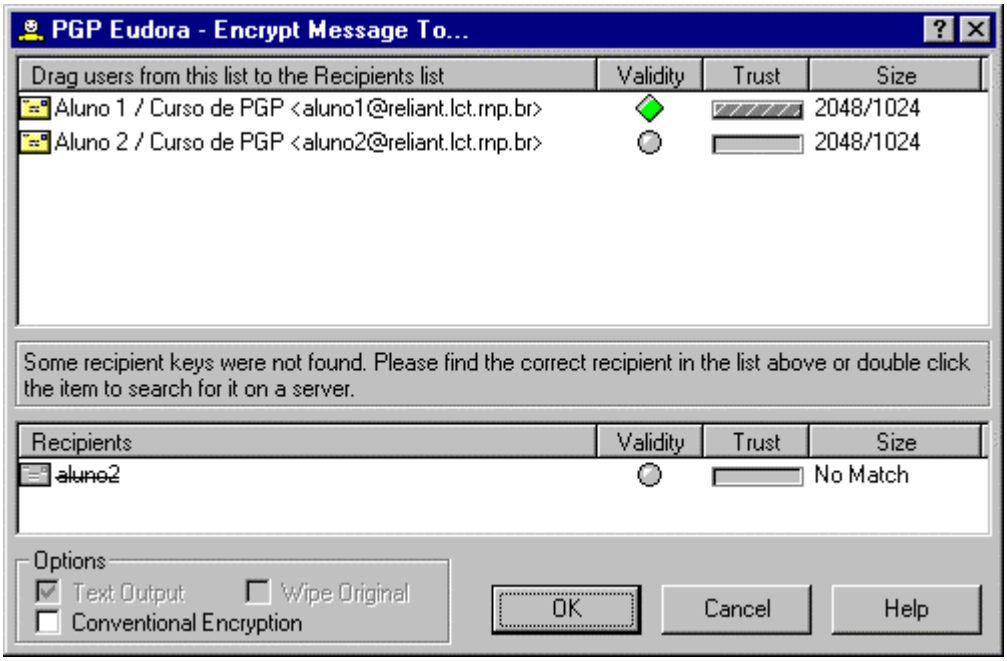
 serve para extrair uma chave de uma mensagem, como fizemos à pouco.

 decripta e/ou verifica assinaturas.

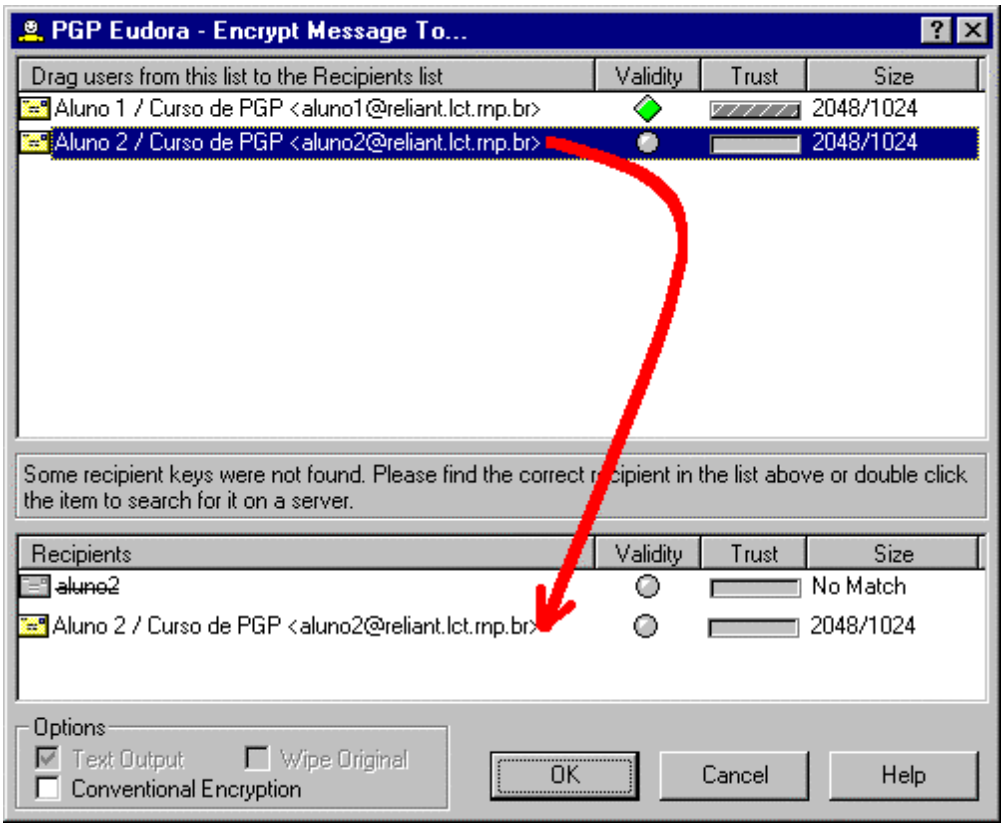
Vamos, a título de exemplo, enviar uma mensagem encriptada para Aluno 2.



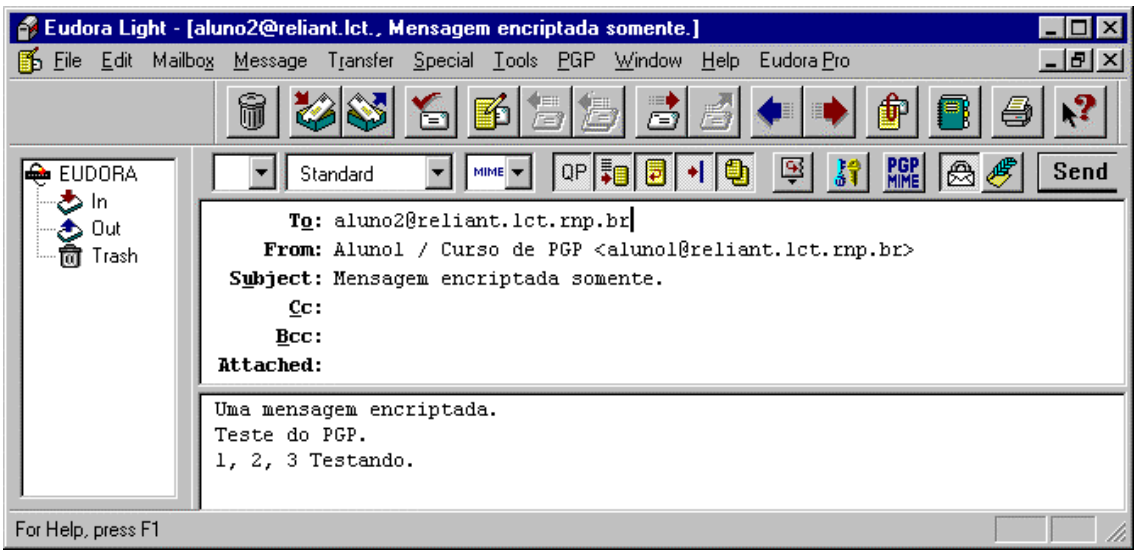
Ao clicarmos *Send*:



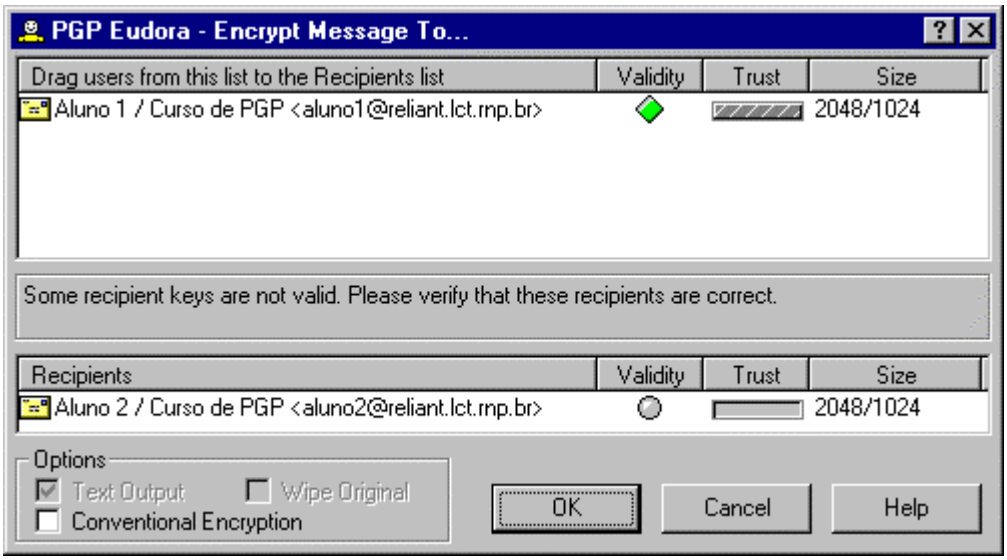
Note na janela *Recipients*: não foi encontrado o usuário aluno2.
Duas formas de contornar o problema:
Arrastamos Aluno 2 da nossa lista de *Recipients* (janela superior).



Corrigimos o endereço no Eudora (de *aluno2* para *aluno2@reliant.lct.rnp.br*) para a forma que se encontra no *KeyRing*.

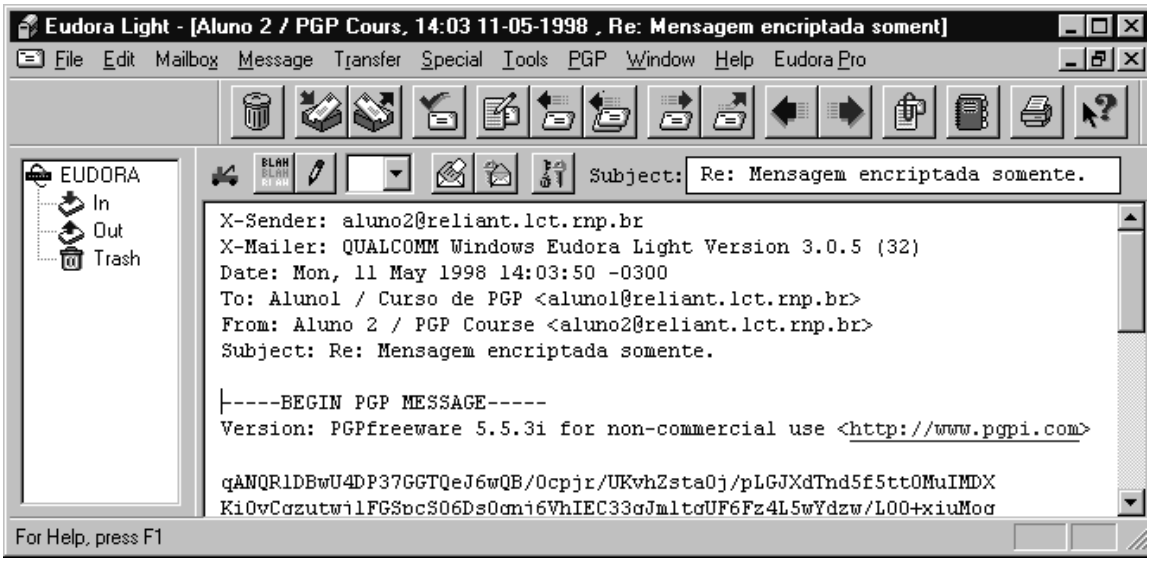


Dessa forma, a chave pública é encontrada no *KeyRing*.



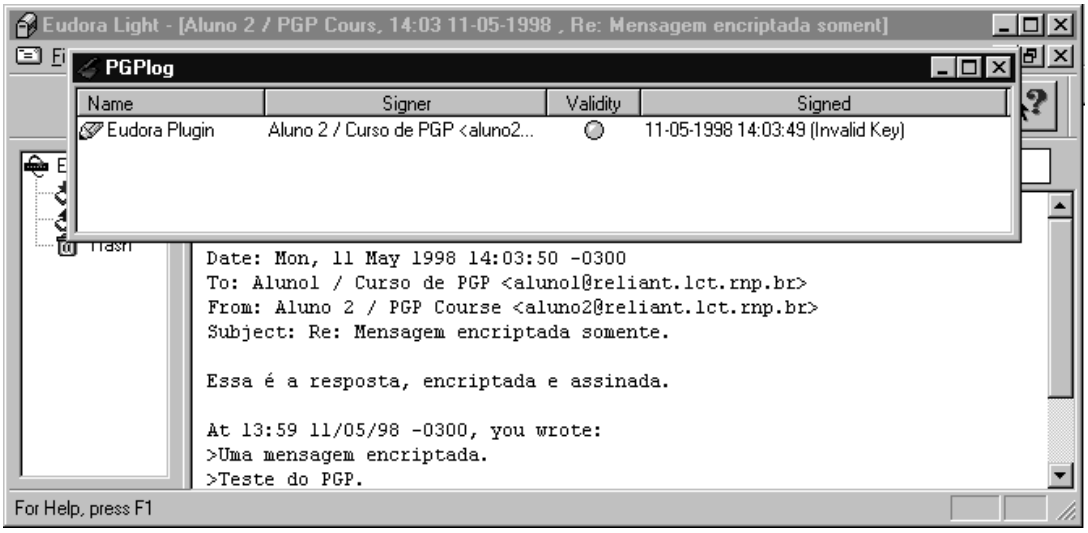
O que fizemos com *aluno2@reliant.lct.rnp.br* pode ser repetido para qualquer número de usuários (caso de múltiplos recipientes).
Como já temos a relação de destinatários pretendida, basta clicar *Ok*. O PGP encripta a mensagem e o Eudora despacha-a pela rede.
Vejamos o resultado na cópia da mensagem que foi para o nosso *OutBox*.

Agora, nosso destinatário nos responde a mensagem de forma encriptada e assinada.

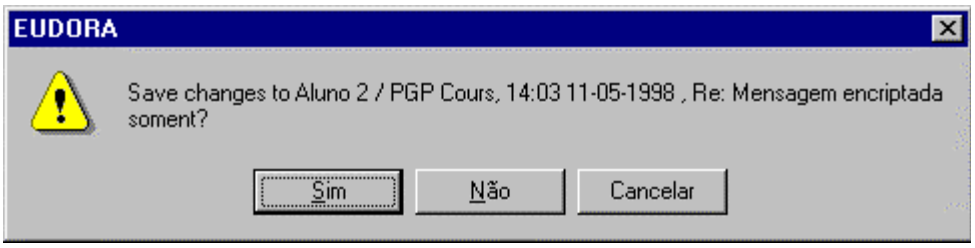


Para decriptá-la, clicanos no ícone da mensagem saindo do envelopinho - *Decrypt PGP Encrypted Email Message*  .
O PGP pede a frase-senha parar recuperar nossa chave privada.

Note a janelinha PGPlog mostrando a autenticidade da mensagem (verificação da assinatura). Quanto a não-validade da chave, vide Capítulo, Item, Subitem para como tornar a chave válida.



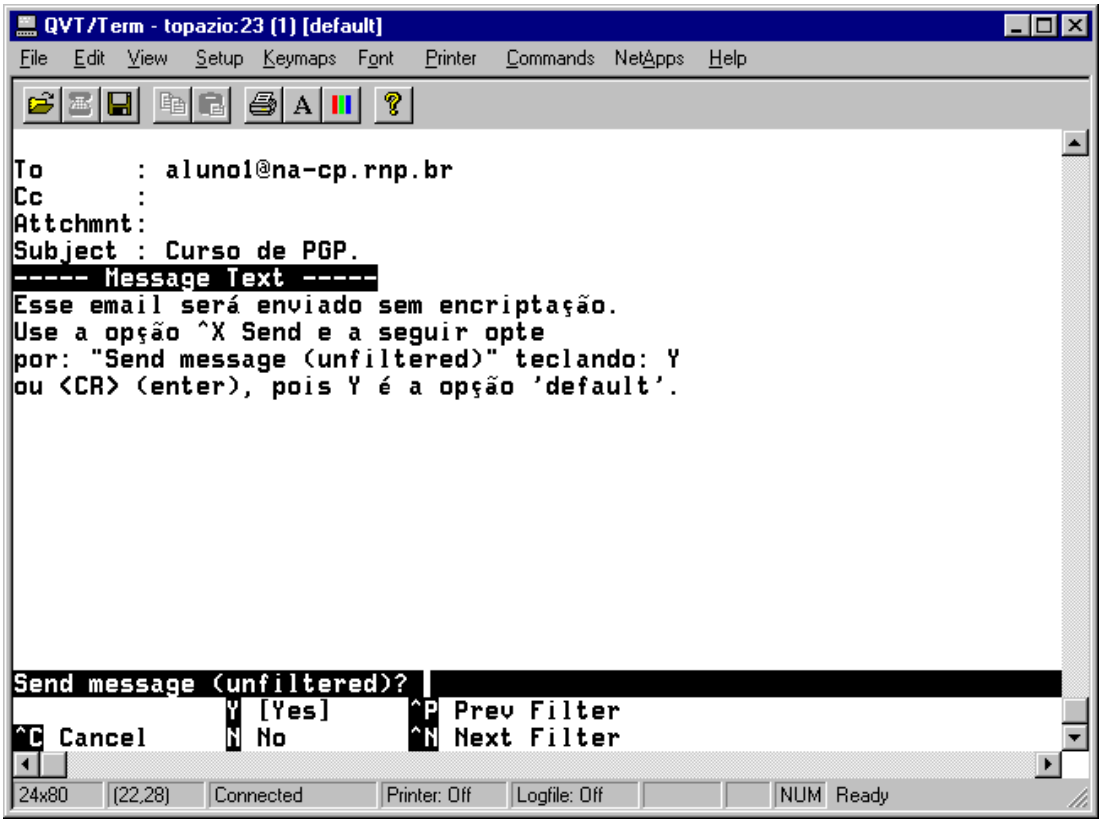
Se quisermos salvar a mensagem decriptada em nossos folder, basta fechar a mensagem e respoder *Sim* a caixa de diálogo que irá surgir. Caso clicarmos a opção *Não* a mensagem será mantida encriptada no Folder *In*.



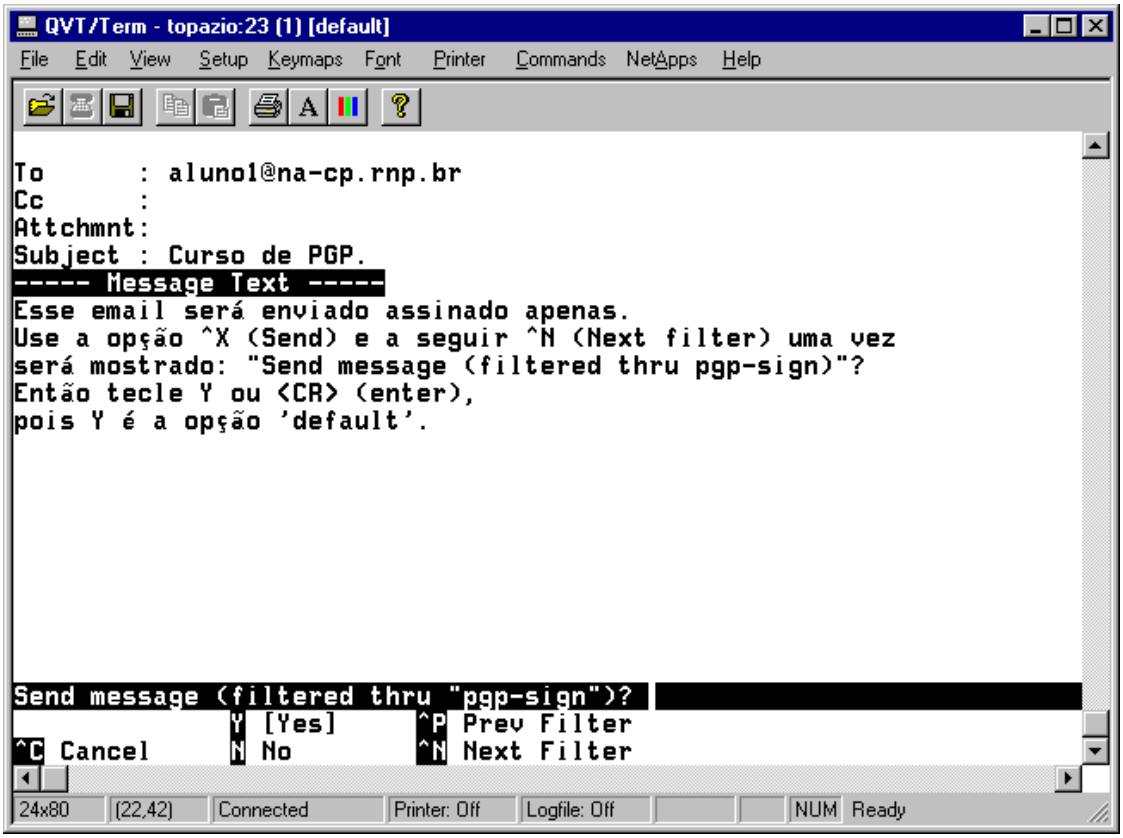
Para leitura de vários e-mails encriptados em seqüência, o PGP do Windows oferece o recurso de 'cache' de frase-senha. Com isso não precisamos digitar a frase-senha a cada mensagem, como fazemos com o Pine. Para personalizar esse item, escolha *Preferences* no menu *PGP* do Eudora, e partir de lá, escolha *General*. Vide Capítulo 5, peculiaridades das versões do PGP.

4.2 Integrado ao Pine

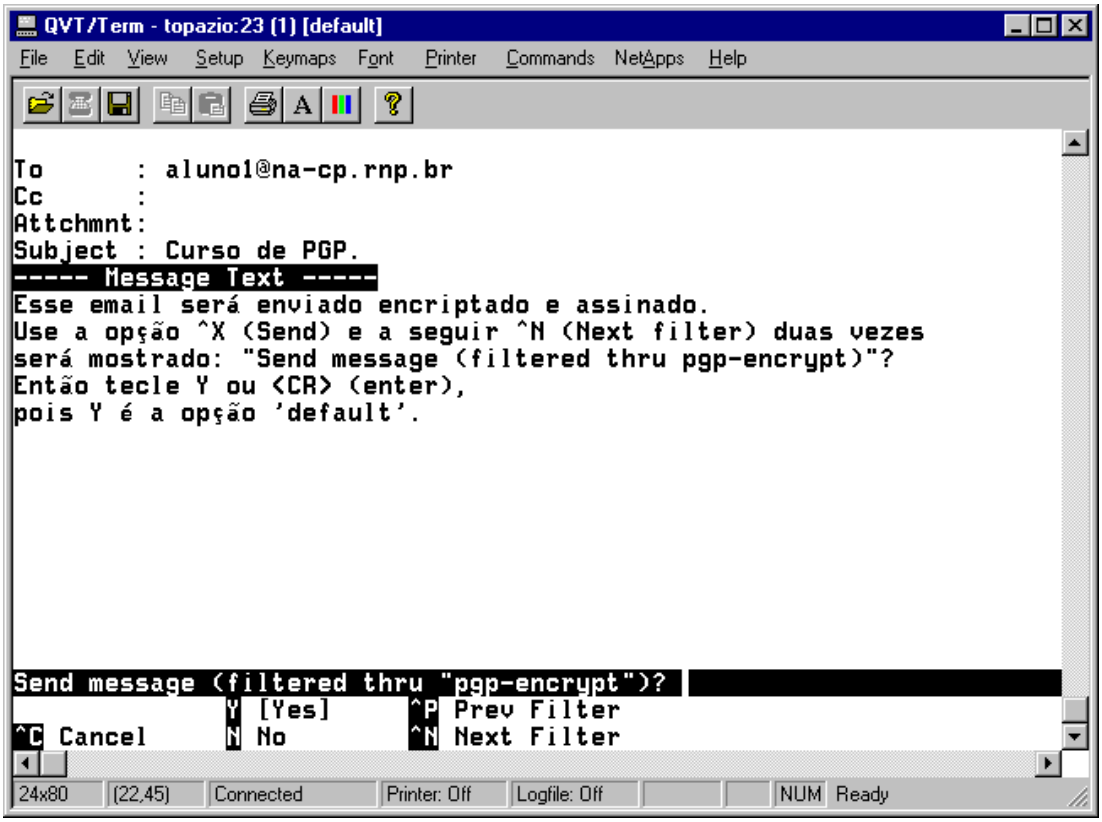
A integração do PGP ao Pine foi feita através de um script *csh* que implementa o PGP como filtro do Pine. O script está localizado no diretório */usr/local/bin* das máquinas Unix do núcleo de Campinas e se chama: *pine-pgp*.
Para mandar um e-mail sem encriptar, basta teclar ^X (Control-X) e optar por Unfiltered. Isso é equivalente ao Send de um Pine sem o FiltroPGP. Suponha para fins de estudo de caso que o usuário *drjones@na-cp.rnp.br* está se correspondendo conosco (*aluno1@na-cp.rnp.br*).



Para se enviar uma mensagem assinada, *DrJones* teclaria ^N (Next Filter) uma vez a partir da tela acima. Nesse ponto, já se presume que o usuário *DrJones* tenha criado seu próprio par de chaves e tenha entregue sua chave pública a *aluno1@na-cp.rnp.br* (para que nós possamos nos certificar da autenticidade da mensagem - vide capítulo 2, item 3; *DrJones* deveria ler o capítulo 3, item 1, subitem 1 e *aluno1* o capítulo 3, item 2, subitem 1).

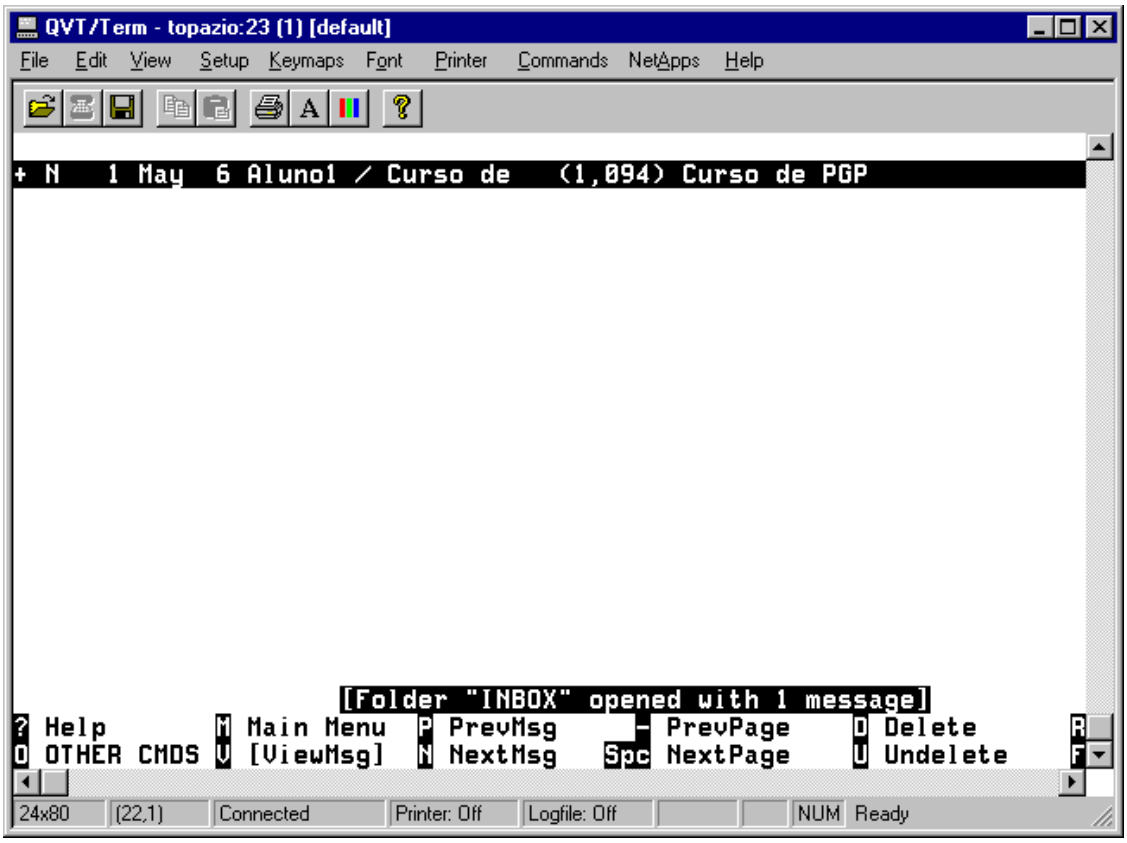


Para se enviar uma mensagem encriptada e assinada, opte pelo filtro seguinte. Nesse ponto presume-se que *aluno1@na-cp.rnp.br* tenha gerado seu par e passado uma cópia de sua chave pública para *DrJones*. (vide capítulo 2, itens 1 e 2; *aluno1* deve procurar o capítulo 3, item 1, subitem 3 e *DrJones* capítulo 3, ítem 2, subitem 1). A versão de filtro do Pine não possibilita gerar uma mensagem encriptada somente.

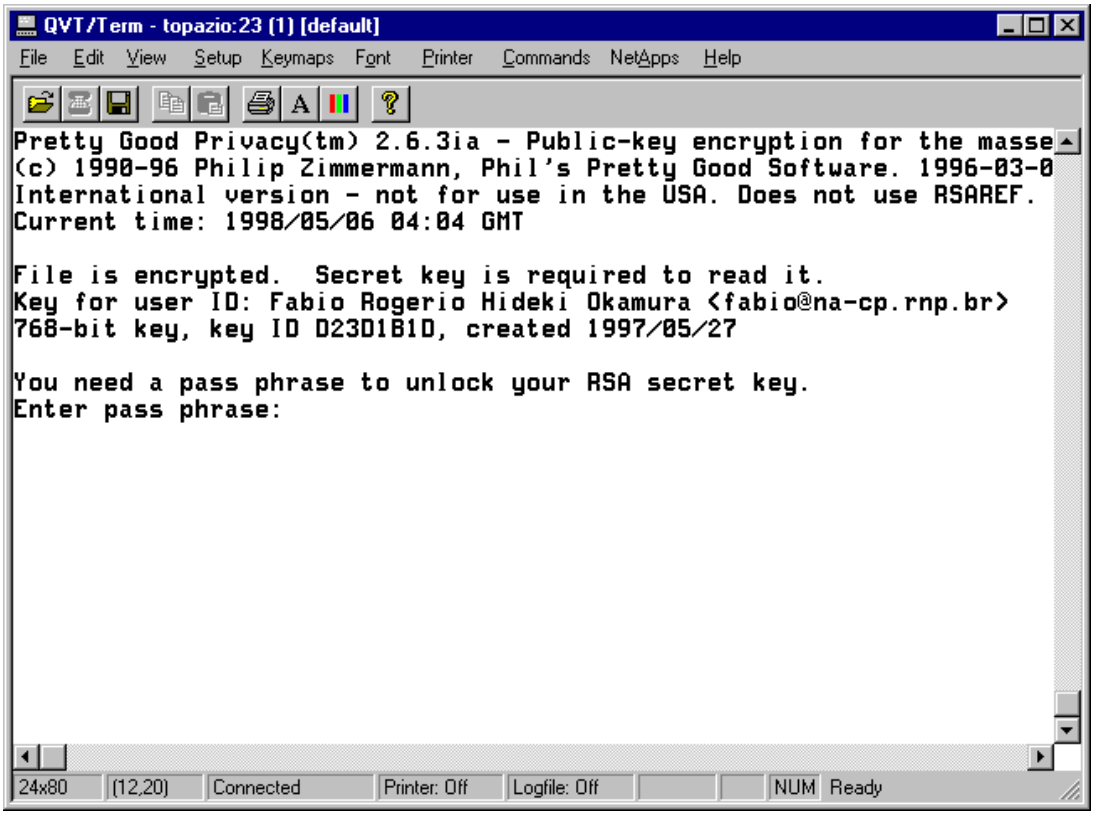


Se teclar ^N (Next Filter) mais uma vez a partir da situação mostrada acima, retornaremos a opção *Send message (unfiltered)*, ou seja, sem passar pelo FiltroPGP. É um ciclo. Podemos retornar sempre a situação anterior via ^P (Previous Filter).

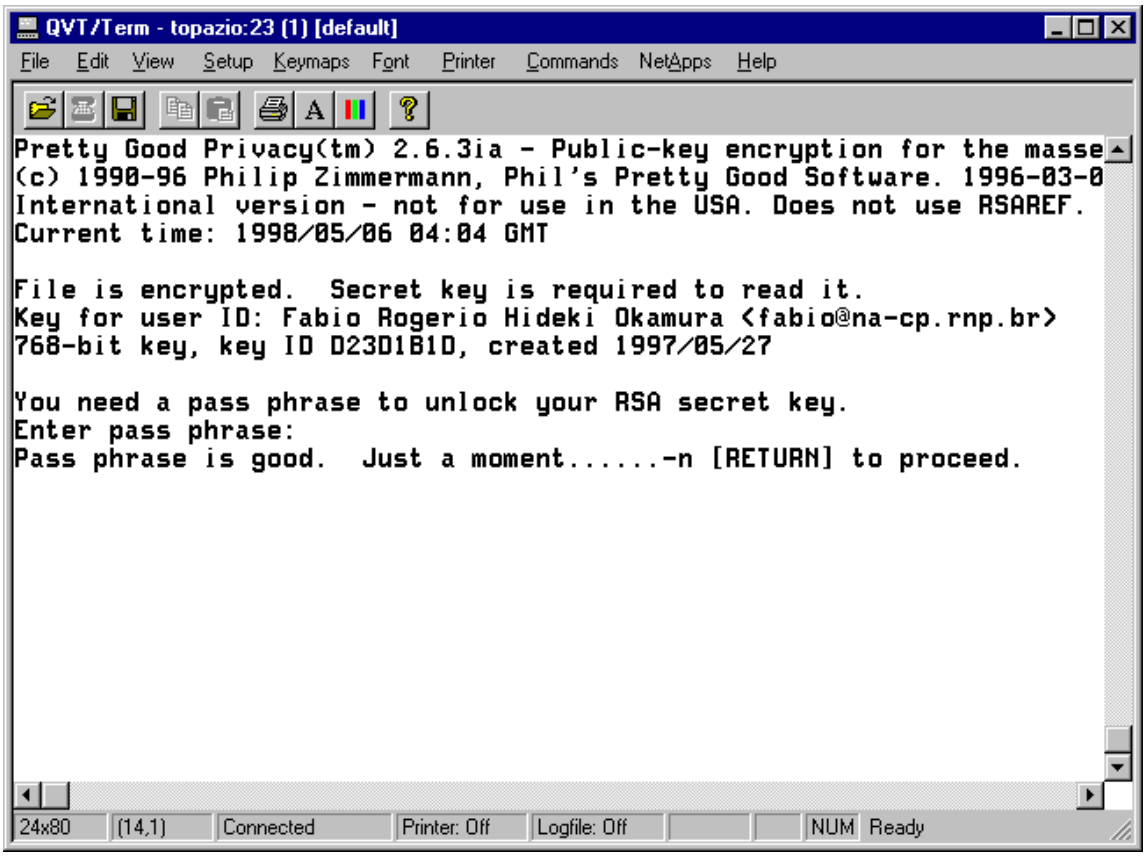
Suponha agora que tenhamos enviado uma mensagem encriptada para o usuário *fabio@na-cp.rnp.br* (temos a chave pública desse usuário).



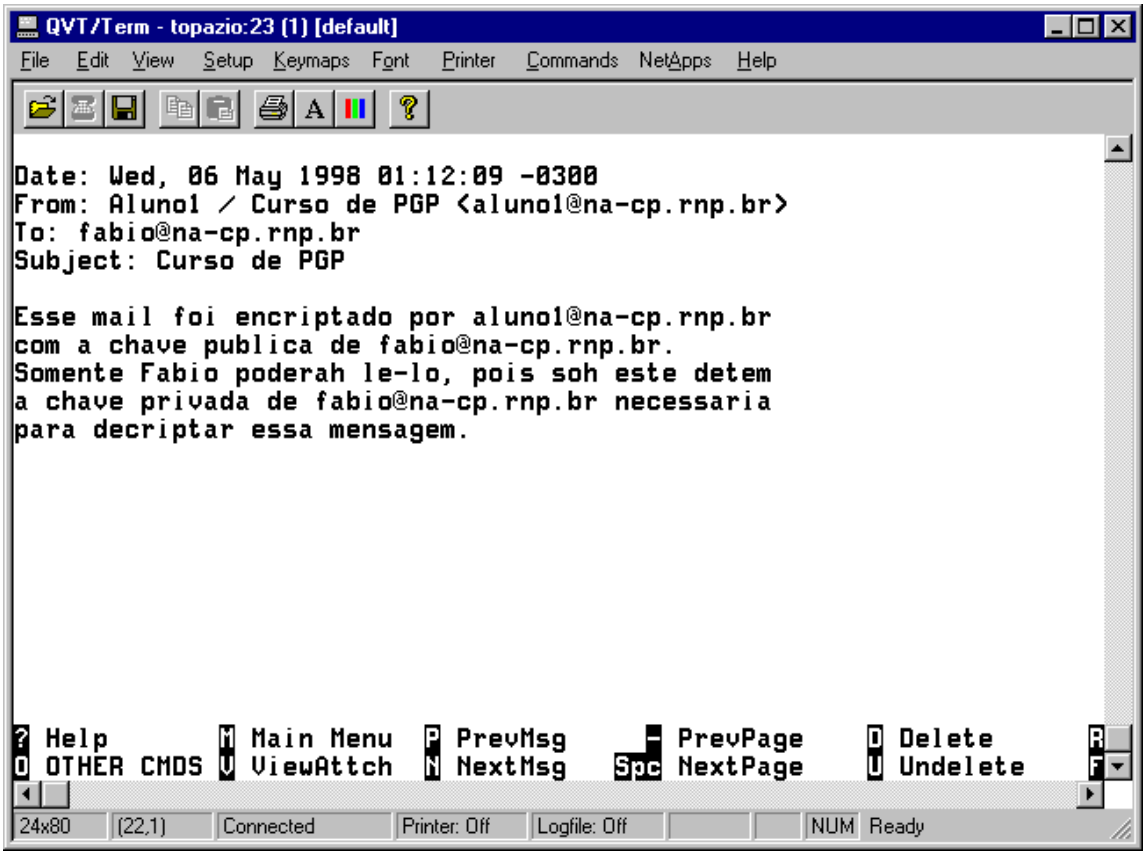
Quando Fábio abre a mensagem, o fluxo do conteúdo do e-mail é passado ao filtro que dispara o pgp:



e depois que é entrada a frase-senha:



Quando for teclado <CR> (enter, ou return):



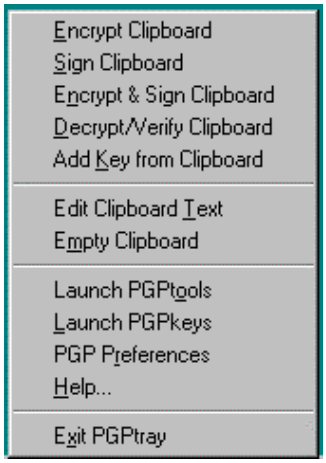
Para preservarmos a mensagem criptografada, devemos optar por *Save* se quisermos preservar a mensagem descriptografada, optamos por *Export* ou *Save attachment* (vide curso de Pine).

5. Peculiaridades das versões do PGP

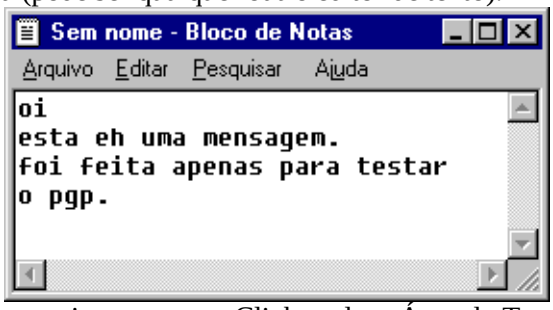
Uma das peculiaridades do PGPv2.6.3i que deixou de existir nos PGPs mais recentes é poder usar a 'flag' (=opção) *-a* como opção 'stand-alone' (=sem estar ligada a encriptação/decriptação/autenticação). Essa opção possibilita tornar textual qualquer tipo de arquivo, mesmo os já passíveis de leitura. Despontando como uma opção melhor ao próprio uuencode do Unix. Quando entramos com o comando: *pgp -a mensa.txt*, o PGP comprime o arquivo *mensa.txt*, depois aplica o algoritmo RADIX64 (torna a mensagem imprimível, usando somente caracteres ASCII com até 7 bits) resultando num arquivo *.asc* passável por email. Tudo que o recipiente tem que fazer para restaurar o arquivo é *pgp -a mensa.txt.asc*. O arquivo *.asc* é na maioria das vezes 2/3 do tamanho do original, pois dos 50% conseguido na compressão, 25% é acrescentado na conversão de binário->ASCII.

Todo PGP versão UNIX faz uso de 'pipes'. De sorte que os erros são passados para o 'handle' *stderr* e a saída encriptada para o *stdout*. Isso facilita a integração do pgp aos clientes de e-mail do Unix. Tudo que se tem a fazer é direcionar o fluxo de caracteres do e-mail para o PGP, este processa a decriptação/encriptação/autenticação e joga para a saída; o cliente então captura essa saída, como vimos no caso do Pine.

Já o PGP do Windows, utiliza o esquema de transferência de dados do Windows: Clipboard (=Área de Transferência). Como se pode ver pelos 7 itens de menu do PGPTray.

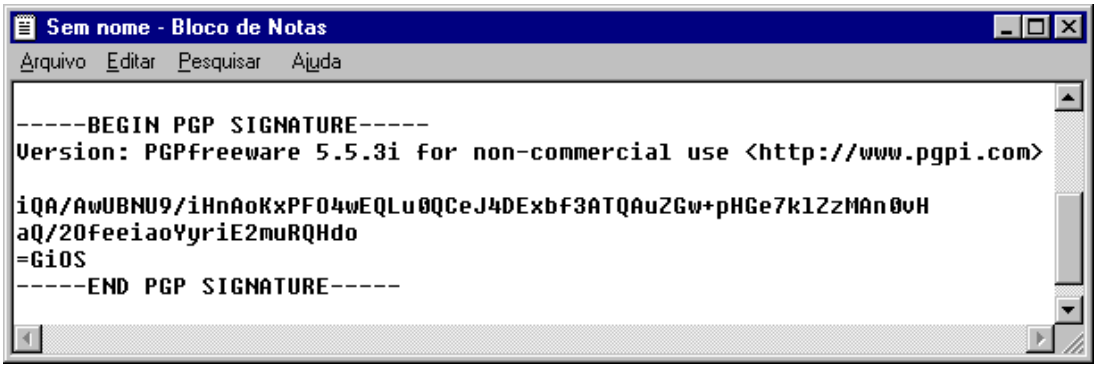


Vamos simular um uso: Digitamos um texto qualquer no Notepad ou Bloco de Notas, como queira (pode ser qualquer outro editor de texto).

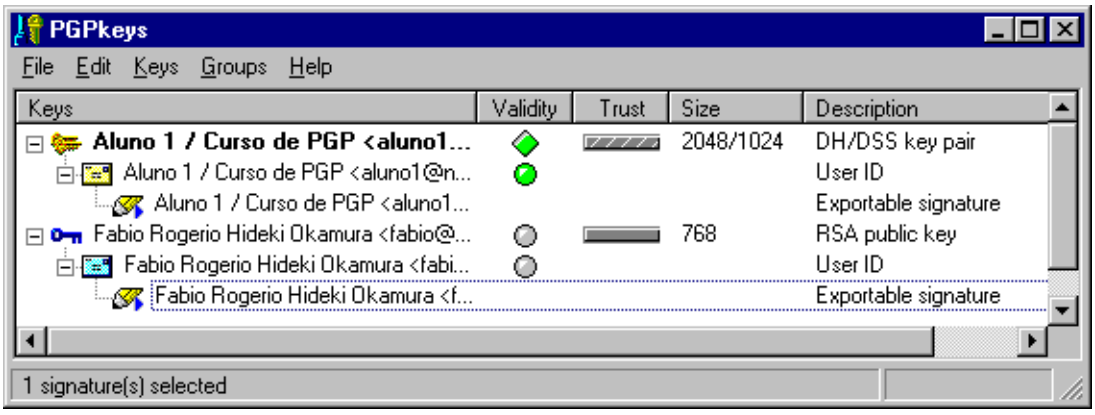


marcamos o texto e copiamos para o Clipboard ou Área de Transferência (*Editar->Copiar)*.

Agora vamos por exemplo, assinar o Clipboard. Abra o PGPTray com um clique no seu ícone e escolha a opção: *Sign Clipboard*. Ele apresentará a tela, velha conhecida nossa, pedindo a chave com a qual iremos assinar e a frase-senha respectiva (Vide assinando mensagens com o PGP, capítulo 3, item 4). Depois de feito o processamento, nós retornamos ao Bloco de Notas e usamos a opção: *Editar->Colar* (Paste). E voilá:



Podemos fazer essa operação de dentro de um cliente de e-mail, como o Eudora, ou qualquer outro de nossa preferência. Isso torna o PGP compatível com virtualmente todo cliente de e-mail que possa existir ou vir a existir num futuro próximo!
Outra particularidade desse PGP, o leitor já deve ter percebido. É a facilidade própria do ambiente gráfico em relação aos comandos de linha. Todas as informações já estão na mão como na janela de abaixo:



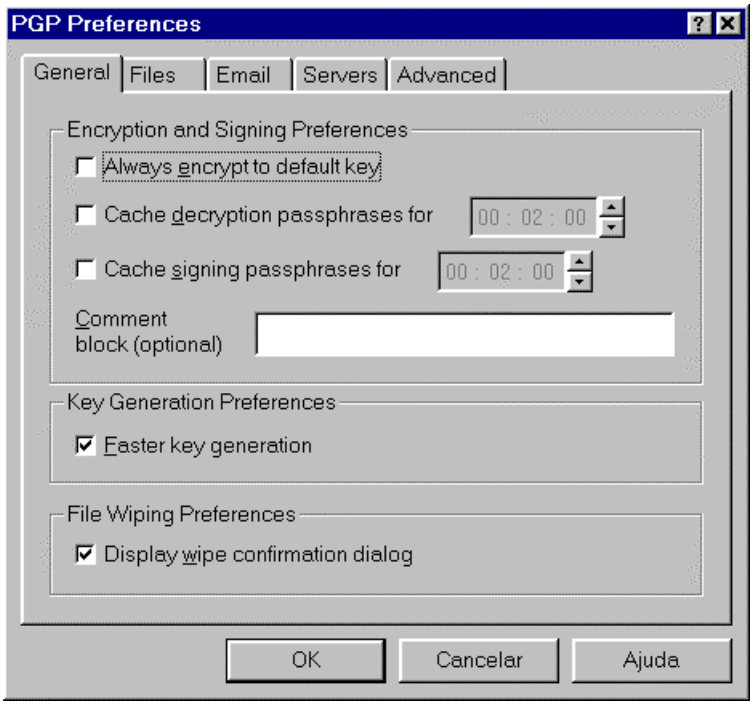
Duo de Chaves, indicam par (secreta+pública). Uma chave só, pública - como podemos ver em descrição.

Chaves amarelas significam, algoritmo DH/DSS; azuis RSA. O nível de confiança (trust) é gráfico (no exemplo acima temos aluno1 com máxima confiança e Fábio com confiança nula). A chave default é marcada com um losango. Uma chave expirada é marcada por um relógio. Bolinha cinza é uma chave inválida (aquela que não recebeu nossa assinatura). O envelopinho é o ID (nome+email). O lápis é a assinatura. Além de outras informações como tamanho da chave e outras passíveis de serem habilitadas via: *Keys->Select Columns*. Veja o manual que acompanha o software para maiores detalhes.

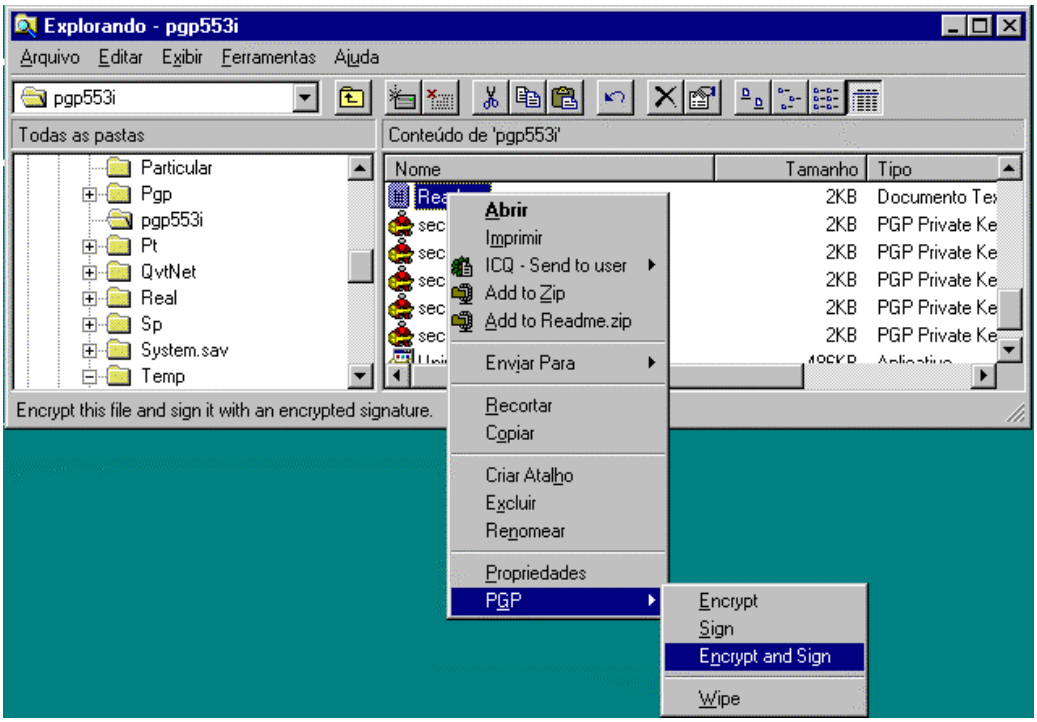
Podemos customizar o PGP do Windows via *Edit->Preferences*, bem diferente das versões comando de linha, customizáveis por arquivos de configurações gerados no diretório do usuário.

Aliás, uma característica interessante introduzida na versão Windows é a cache de frase-senha. O PGP memoriza por um tempo especificado nossa frase senha, útil quando recebemos muitos e-mails e queremos evitar ficar teclando a frase senha para cada e-mail a ser lido. Esse é acessível via *Edit->Preferences->Cache Decription Passphrase for*. Quando o usuário experimentar o PGP com Pine vai

desejar imensamente esse recurso...



O PGP se encontra altamente integrado ao ambiente Windows, como podemos ver no caso do gerenciador de arquivos do Windows (Explorer). Navegamos até o arquivo desejado (ReadMe.txt nesse caso), e clicamos com o botão direito. Escolhemos *PGP* para acessar as funções do PGP. No exemplo mostrado abaixo, estamos optando por encriptar assinado. Se fizermos a mesma operação num arquivo que contenha uma chave o PGP perguntará se queremos importá-la ao nosso anel.



Existem outros inúmeros 'features', o leitor deve consultar o manual para melhor usufruto do PGP. Nossa missão era fornecer um 'bird´s eye view'.

6. Considerações Finais

Como diz Daniel Wey, pesquisador do LSI/USP e consultor da Internet World, "PGP é o software mais utilizado em todo o mundo para encriptação de mensagens de e-mail, por oferecer duas vantagens que não estão disponíveis em praticamente nenhum outro software do gênero: algoritmo de encriptação forte, que torna suas mensagens praticamente invioláveis, e o fato de estar disponível gratuitamente, inclusive com o código fonte".

E ao que parece vai continuar na frente, no momento em que estamos escrevendo essas considerações, a Network Associates está para distribuir o novo PGP (a versão 6.0), entre os novos 'features' temos: *designated revokers*, você agora pode autorizar um amigo em que você confia para 'dizer' para o mundo para não confiar mais na sua chave, caso você a perca ou for roubada; *photo IDs*, pequenos arquivos gráficos atados às chaves para segurança adicional.

Para os curiosos de plantão, colocamos nas referências as RFCs que contêm os algoritmos usados pelo PGP.

Esse guia, nem de leve explora todas as potencialidades desse software. Seria interessante que o leitor procurasse dar uma olhadinha nos manuais que acompanham o software (referências 1, 2 e 3).

Para os que não são muito afeitos ao inglês, o guia de José Daniel Ramos Wey publicado na Internet World (referência 4) é uma boa pedida.

Uma dúvida freqüente ao curso que originou essa apostila é o por quê do uso do PGP no núcleo da RNP de Campinas. Respondo fazendo minhas as palavras-metas do PGP: segurança, privacidade, autenticidade com conforto e simplicidade.
Boa sorte a todos, e continuem encriptando!!!

7. Referências

1. Stallings, William. *Protect Your Privacy - The PGP User's Guide*. Prentice Hall, 1995.
2. Zimmermann, Philip. *Pretty Good (tm) Privacy - Public Key Encription for the Masses - PGP (tm) User´s Guide Vol I (Essential Topics) and Vol II (Special Topics)*. Pretty Good Software, 1994
3. Network Associates Inc. *PGP for Personal Privacy Version 5.5 For Win95/NT - User´s Guide.*Network Associates Inc., 1998
4. Wey, D., "*Criptografia bem amigável*", Internet World, Fevereiro 1998, págs 80-85.
5. RFC 1991 Atkins, D., "*PGP Message Exchange Formats*", RFC 1991, August, 1996.
6. RFC 1321 Rivest, R., *"The MD5 Message-Digest Algorithm"*, RFC 1321, April 1992.
7. RFC 2144 Adams, C., "*The CAST-128 Encryption Algorythm*", RFC 2144, May 1997.
8. RFC 2313 Kaliski, B. "*PKCS #1: RSA Encryption Version 1.5"*, RFC 2313, March 1998.
9. RFC 1421 Linn, J., "*Privacy Enhancement for Internet Electronic Mail: Part I: Message Encryption and Authentication Procedures*", RFC 1421, February 1993.
10. RFC 1422 Kent, S., "*Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management*", RFC 1422, February 1993.
11. RFC 1423 Balenson, D., "*Privacy Enhancement for Internet Electronic Mail: Part III: Algorithms, Modes, and Identifiers*", RFC 1423, February 1993.
12. RFC 2045 Freed N. & Borenstein N., "*Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*", RFC 2045, November 1996.
13. RFC 2046 Freed N. & Borenstein N., "*Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*", RFC 2046, November 1996.
14. RFC 2047 Moore K., "*Multipurpose Internet Mail Extensions (MIME) Part Three: Header Extensions for Non-ASCII Text*", RFC 2047, November 1996.
15. RFC 2048 Freed N., Klensin J., Postel J., "*Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures*", November 1996.
16. RFC 2049 Freed N. & Borenstein N., "*Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples*", RFC 2049, November 1996.