

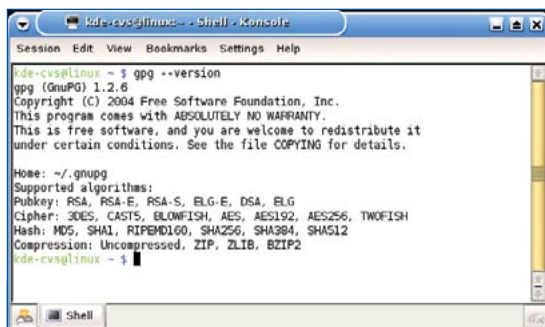
Criptografando mensagens de email no KMail, Mozilla Thunderbird e Evolution

# Assinada, selada e despachada

Para que suas missivas digitais sejam entregues sem que o carteiro virtual dê uma olhada, a melhor pedida é criptografar tudo. Este artigo descreve como usar os recursos de cifragem já existentes nos manjadíssimos Thunderbird, Kmail e Evolution.

POR FRAUKE OSTER

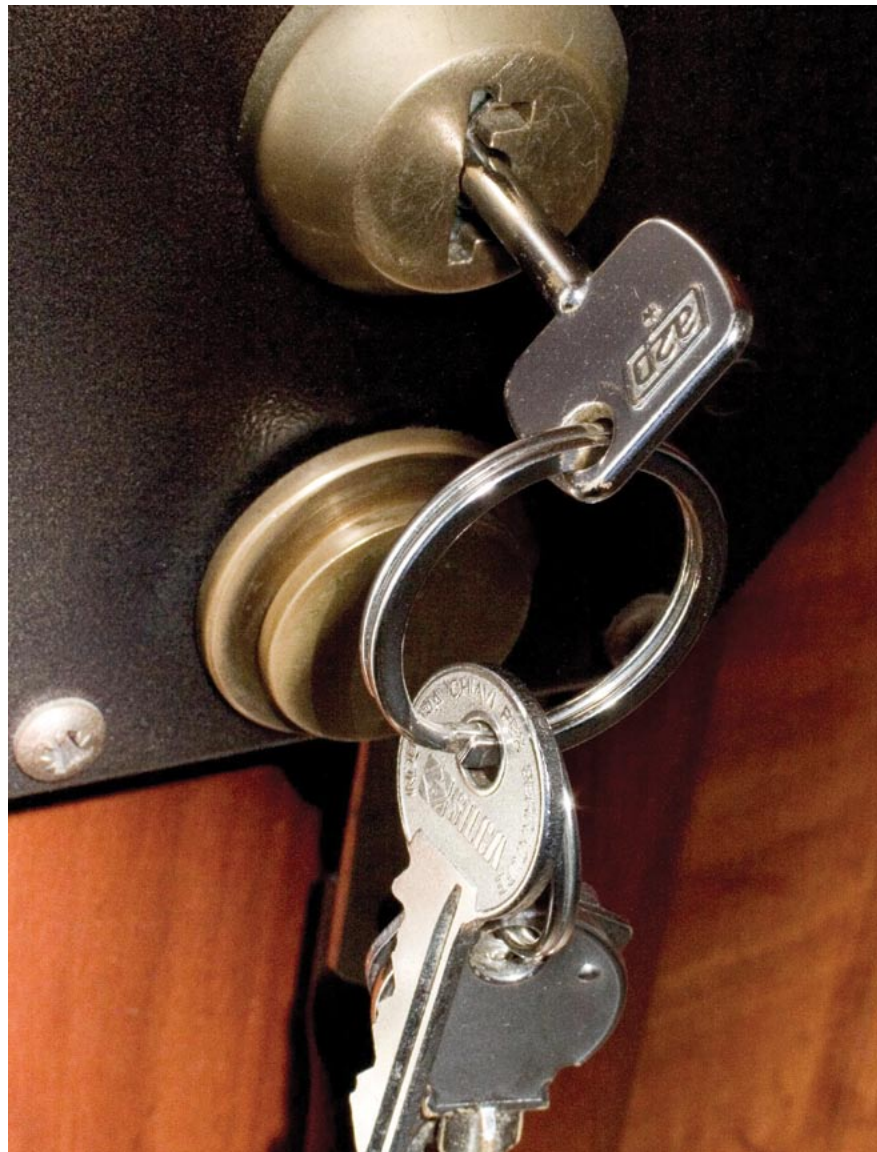
Os falsários de todos os tempos devem ter inveja dos *spoofers* de hoje em dia. Nestes dias muito estranhos, a Internet torna o ofício da impostura uma tarefa bastante simples e difícil de desmascarar. Não é necessário forjar uma assinatura para despachar um email em nome de outrem: basta manipular corretamente as informações contidas no cabeçalho, especialmente o campo *From* (“De” ou “Remetente”, dependendo da tradução). Os protocolos usados para o serviço de correio eletrônico também não colaboram nem um pouco para prevenir esse tipo de contrafação. Se você quiser que as pessoas para quem você escreve sejam capazes de determinar a autenticidade de suas mensagens e impedir que rufiões coloquem palavras em sua boca, cultive o saudável hábito de assinar digitalmente suas cartas. O mesmo pode se dizer da criptografia – ou você realmente quer que algum administrador abelhudo saiba de seus



```
kde-cvs@linux: ~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
kde-cvs@linux ~ $ gpg --version
gpg (GnuPG) 1.2.6
Copyright (C) 2004 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.

Home: ~/.gnupg
Supported algorithms:
Pubkey: RSA, RSA-E, RSA-S, ELG-E, DSA, ELG
Cipher: 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH
Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512
Compression: Uncompressed, ZIP, ZLIB, BZIP2
kde-cvs@linux ~ $
```

**Figura 1:** Muitos programas de email precisam do utilitário *gpg* para criptografia e assinatura. `gpg -version` nos diz qual a versão instalada no sistema.



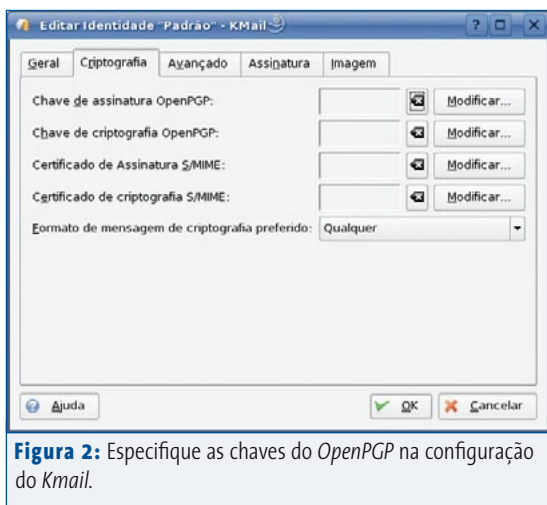


Figura 2: Especifique as chaves do OpenPGP na configuração do Kmail.

assuntos particulares? Qualquer pessoa que tenha acesso a um dos servidores intermediários entre sua caixa de saída e a caixa de entrada do destinatário poderia ler tudo o que você escreveu.

O GNU Privacy Guard (GnuPG) [1] é um programa que protege seus emails contra monitoração e manipulação. O GnuPG é um sistema de criptografia que usa chaves assimétricas. Para o usuário, isso significa possuir duas chaves, uma pública e outra privada – geradas simultaneamente e conhecidas como *par de chaves* (*key pair*). A chave privada é mantida em segredo com uma senha. Com ela, você descriptografa mensagens e as assina.

Por outro lado, a chave pública tem esse nome porque tem que, obrigatoriamente, ser distribuída. Todos os destinatários para os quais você escreve devem possuir uma cópia de sua chave pública. Com a chave pública, seus contatos podem criptografar mensagens antes de mandá-las a você. Note que a chave pública serve apenas para criptografar: a única maneira de descriptografar essas mensagens é com sua chave privada. A chave pública serve também para que seus contatos consigam verificar se a mensagem veio mesmo de você – ou seja, é usada para verificar a autenticidade de sua assinatura digital. Quando você assina uma mensagem, o GnuPG usa sua chave secreta para gerar

um *hash* do texto útil (“corpo do email”) e cria um anexo com ele. O destinatário usa a chave pública para, decodificando sua assinatura, conferir se você é você mesmo.

Para poder proteger suas comunicações, entretanto, você precisa de duas coisas: do programa GnuPG e de um cliente de email que trabalhe com ele. Neste artigo, nos concentraremos nos “campeões de audiência” dessa seara: KMail, Thunderbird e Evolution. O comando `gpg -version` informa se o GnuPG já está instalado em seu sistema e, se estiver, indica qual a versão. (figura 1). Se o comando cuspir uma mensagem de erro, será preciso instalar o GnuPG a partir dos CDs de sua distribuição. O pacote é normalmente chamado de *gpg* ou *gnupg*; no SUSE 10.0, o pacote está na versão 1.4.2.

Como nem todos os clientes de email são capazes de gerar, eles mesmos, um par de chaves, veremos aqui como gerá-las usando a linha de comando – o que garante que possamos seguir estas ins-

truções em praticamente qualquer sistema. Digite `gpg -gen-key` para chamar o diálogo de geração de chaves. A primeira coisa que o GnuPG vai perguntar é qual mecanismo de criptografia usar. Há três opções, mas a que vem pré-definida – *ElGamal* e *DSA* – é a melhor escolha para começar. Pressione **[Enter]** para confirmar. Podemos então especificar o comprimento da chave. Aqui temos que escolher entre mais segurança e mais desempenho. Uma chave pequena é mais fácil de ser quebrada por malfetores, mas requer menos processamento. O padrão do programa (1024 bits) é o suficiente para a maioria das aplicações. Pressione **[Enter]** mais uma vez para aceitá-lo.

O GnuPG pergunta, então, qual a data de validade (“expiração”) desejada para o par de chaves. Se você quer ter uma rede de confiança meio grande (ver quadro 1), não seria lá muito bom que suas chaves tenham um tempo de vida demasiado curto, já que isso implica em enviar novas chaves para todo mundo sempre que elas expirarem – e esperar que todo mundo cadastre as novas chaves. Se estiver em dúvida, não defina tempo de vida algum.

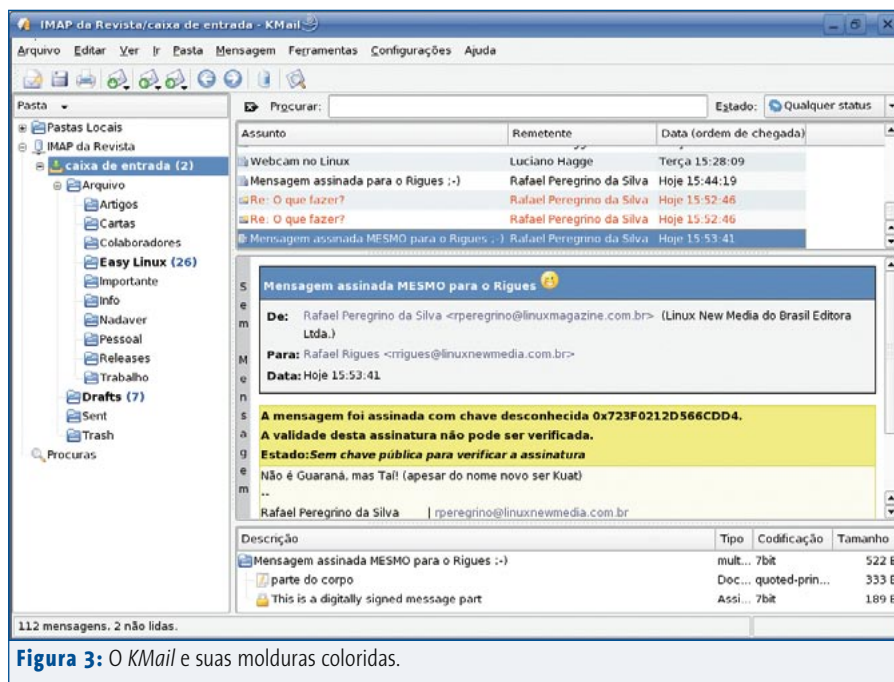
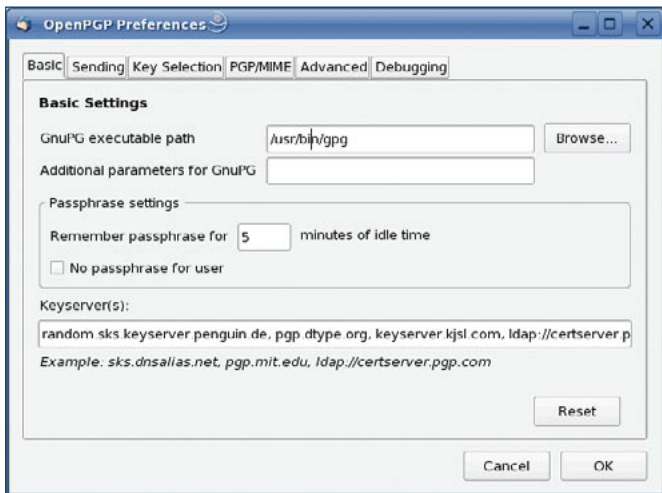


Figura 3: O KMail e suas molduras coloridas.



**Figura 4:** Antes de poder usar o plugin *Enigmail* com seu *Thunderbird*, especifique o caminho para o *GnuPG*.

Se quiser definir uma data de validade, entretanto, será preciso um “certificado de revogação” (*revocation certificate*) para cancelar as chaves antes que a validade vença e removê-las dos servidores de chaves. A melhor opção é, na verdade, criar um certificado de revogação imediatamente após a criação do par de chaves – digite `gpg -output revoke.asc --gen-revoke key-ID` para isso – e o guarde com carinho, em local fresco, seco e ao abrigo de luz, para uso futuro. Informe seu endereço de email como identificador para as chaves (*key ID*). Depois disso, pressione `[y]` para confirmar a data de validade do par de chaves.

Com ou sem data de validade, a próxima etapa é digitar seu nome, um comentário opcional e um endereço válido de email. Esse endereço tem que ser exatamente o mesmo no qual as chaves serão usadas. Pressione, então, `[F]` para finalizar. Na última etapa, o *GnuPG* pede uma frase secreta, que será usada como senha. Como o próprio nome deixa aparente, não basta uma única palavra (embora seja permitido). O ideal é que se digite uma frase completa, com letras, números e, para temperar, caracteres especiais. A segurança do *GnuPG* depende em grande parte de uma frase bem escolhida e difícil. Se alguém conseguir roubar sua chave privada, a frase secreta é a única coisa que vai impedir o salafário sacripanta de decodificar sua correspondência ou de enviar mensagens em seu nome, com assinaturas legítimas.

## Configurando o KMail para usar o GnuPG

Como era de se esperar, as configurações para usar o *GnuPG* no *KMail* (vamos usar a versão 1.8.2) [2] estão em *Configurações* | *Configurar Kmail*, mais precisamente na aba *Ferramentas de Criptografia* dentro do item *Segurança* – mas por enquanto não mexa em nada. Vá até o item *Identidades* e associe a chave que acabou de criar a seu endereço de email: selecione a identidade

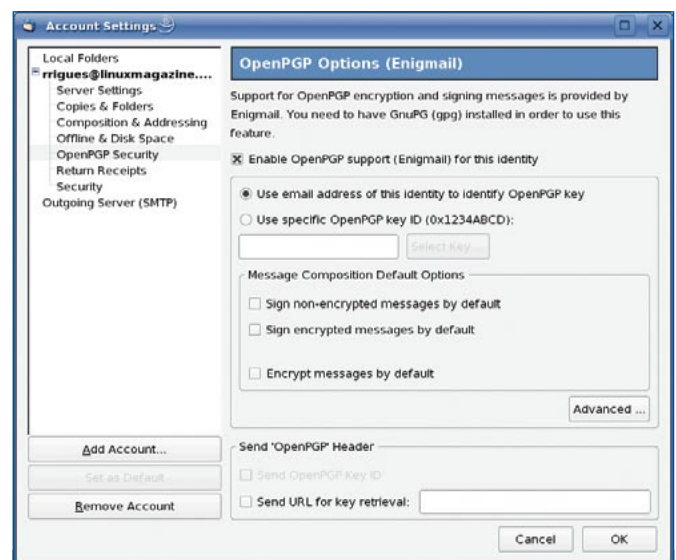
que quer modificar, clique no botão *Modificar* e, na aba *Criptografia*, indique a *Chave de assinatura OpenPGP* e a *Chave de Criptografia OpenPGP* (**figura 2**).

Dois novos botões aparecem na janela de redação de mensagens. O botão com a caneta bico-de-pena é usado para assiná-las; já o do cadeado é para criptografá-las. A opção *Anexar chave pública* do menu *Anexar* permite enviar a sua (ou qualquer outra) chave pública junto com a mensagem.

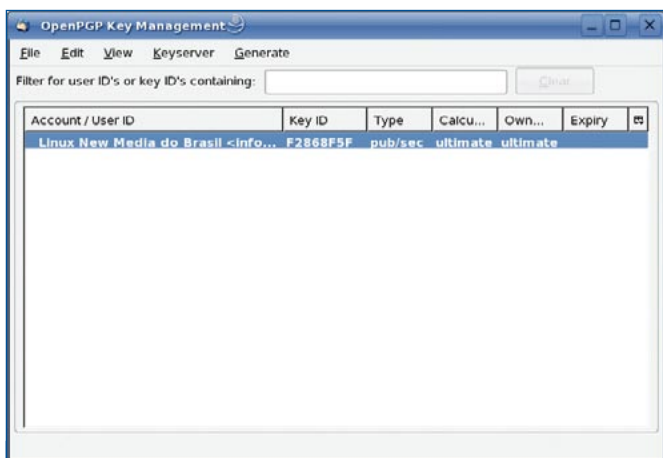
Se você receber de outra pessoa uma mensagem assinada, o *KMail* desenha um quadro em volta da mensagem (**figura 3**). Assinaturas inválidas são “enquadradas” em vermelho. Assinaturas válidas mas nas quais não confiamos (ou seja, não temos sua chave pública) possuem uma moldura amarela. Assinaturas de quem confiamos possuem moldura verde. Com isso, num golpe de vista já podemos distinguir entre mensagens confiáveis e mensagens dúbias. O *KMail* também desenha um quadro azul em volta das mensagens criptografadas que ele conseguir decifrar.

Versões antigas do *KMail* possuem uma desvantagem sem par: o uso da chamada criptografia *inline* – ou seja, o texto da mensagem é cifrado, mas os anexos não. A versão 1.7 e posteriores (a última é a 1.8.2, presente no KDE 3.4.2) adotam o padrão *OpenPGP/MIME*, usada por praticamente todos clientes de email dignos desse nome. O *OpenPGP/MIME* criptografa todos os itens individuais da mensagem, incluindo aí os anexos, e os envia como objetos *MIME* individuais.

As versões do *KMail* anteriores à 1.7 não reconheciam mensagens criptografadas no formato *OpenPGP/MIME* – padrão usado por um sem-número de outros programas. O novo *KMail*



**Figura 5:** As configurações de contas de *Thunderbird* são o lugar certo para ativar o *Enigmail* e especificar as chaves que devem ser usadas.



**Figura 7:** O Thunderbird, com a ajuda de seu fiel escudeiro Enigmail, é o único dos três programas testados que pode criar e manter chaves. Isso significa que os usuários não precisarão abrir um terminal cada vez que uma chave precisar de manutenção.

reconhece ambos os métodos. Se você for um feliz usuário do KMail, migrar para uma versão posterior à 1.7 é uma boa idéia – se você atualiza sempre seu KDE, já deve estar com uma versão bem mais nova que essa. A série 1.6 do KMail acompanhava o KDE 3.2. A série 1.7 veio com o KDE 3.3 e a 1.8 é a nova estrela do KDE 3.4.

Como alternativa, é possível usar o projeto *Ägypten* > [3] para adicionar o suporte ao OpenPGP/MIME no KMail 1.6.2. O SUSE LINUX possui um pacote com um plugin OpenPGP/MIME pronto para usar, mas os usuários de todas as outras distribuições precisam compilá-lo a partir do código fonte. Como para isso será preciso antes compilar e instalar outros seis pacotes que são pré-requisito, atualizar seu sistema para um KDE mais novo talvez seja mais simples e sábio.

## Thunderbird + Enigmail

O Mozilla Thunderbird [4] precisa do plugin *Enigmail* [5] para poder trabalhar com o GnuPG. Baixe o plugin de qualquer um dos mirrors e instale pelo menu *Tools | Advanced (Ferramentas | Avançadas)* do Thunderbird.

É necessário especificar o caminho para o executável GnuPG nas preferências do Enigmail (*OpenPGP | Preferences – figura 4*). Para a maioria das distribuições é `/usr/bin/gpg`. Para os outros campos, os padrões de fábrica devem servir.

O item *OpenPGP Security*, disponível em *Edit | Account settings*, serve para especificar quais chaves o Enigmail deve usar. Primeiro, ative o suporte ao GnuPG clicando em *Enable OpenPGP support (Enigmail) for this identity (figura 5)*. Se a chave for gerada com o endereço de email embutido (como vimos ali atrás), o Thunderbird tentará associar as chaves a suas respectivas contas automaticamente. Se não for esse o caso, escolha a segunda opção

e procure pela chave apropriada na lista. Depois basta informar se o Thunderbird deve assinar e criptografar as mensagens automaticamente, sem a intervenção do usuário.

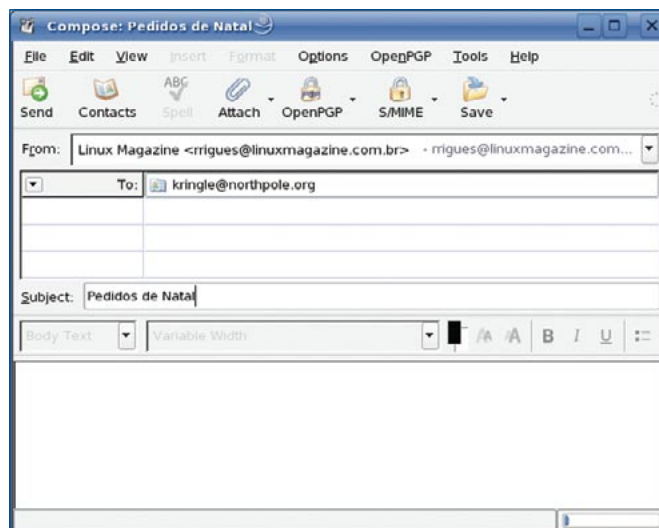
Ao redigir uma mensagem, um botão chamado *OpenPGP* chama a atenção. Ele esconde um menu com várias opções de criptografia e assinatura (**figura 6**). O menu do Enigmail também pode anexar sua chave pública à mensagem. É uma maneira fácil de distribuí-la a seus contatos.

Se receber uma mensagem criptografada ou assinada, o Enigmail realça o cabeçalho logo acima do endereço do remetente. Se a mensagem for assinada, um ícone em forma de pluma será exibido; se for cifrada, aparece uma chave. Um clique nesses ícones revela informações mais completas sobre os mecanismos de assinatura e cifragem – uma maneira fácil de descobrir a origem da mensagem. O Thunderbird lê e escreve nos dois métodos de criptografia que citamos: *inline* e *OpenPGP/MIME*.

A tela sob o menu *OpenPGP | Key Management* possui uma ferramenta bastante útil para administrar suas chaves. Ela pode, por exemplo, listar todas as chaves públicas em seu sistema. É possível, então, assinar as chaves, criar uma nova chave e adicionar usuários àquela chave. Para revogar uma chave, clique com o botão direito do mouse sobre ela na janela *OpenPGP Key Management (figura 7)* e selecione a opção *Revoke Key*. Infelizmente, o Enigmail não dispõe de uma forma fácil de administrar servidores de chaves

## Evolution: simples e seguro

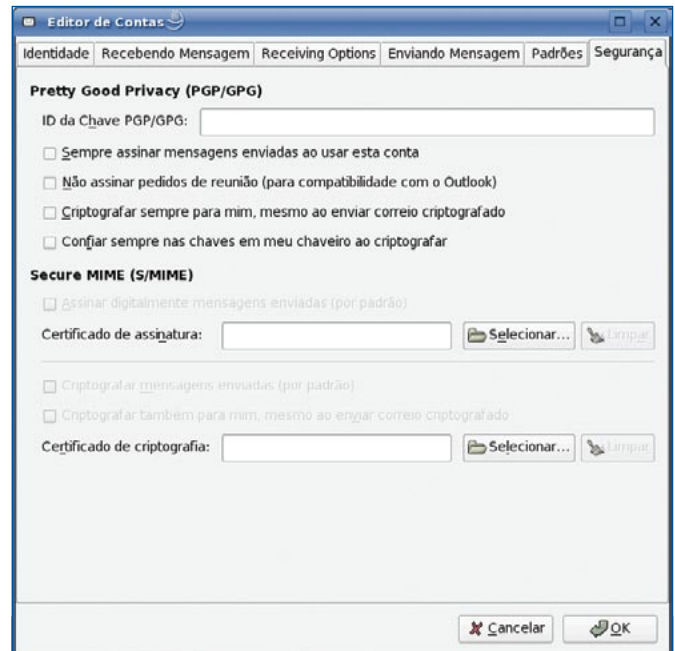
Para usar o GnuPG com a versão 2.4 do Novell Evolution [5], abra a tela de configurações (menu *Editar | Preferências*) e selecione a conta de email para a qual quer definir uma chave



**Figura 6:** Ao instalar o Enigmail, um novo botão chamado *OpenPGP* é mostrado na janela de redigir mensagens.

GnuPG. Clique em *Editar* e informe o identificador da chave (*ID da Chave PGP/GPG*) na aba *Segurança* (figura 8). Para saber o ID de cada chave, abra um terminal e digite o comando `gpg -list-keys você@seudominio.com`. O GnuPG mostrará a abrangência da chave (por exemplo, *pub* para chaves públicas), o comprimento e o tipo de criptografia (por exemplo, *1024D* para uma chave DSA de 1024 bits), o identificador associado (*key ID*) e as datas de criação e validade. O Evolution também permite que, em vez de informar o *key ID* se diga o endereço de email correspondente.

Na mesma tela podemos especificar que queremos assinar todas as mensagens enviadas de agora em diante, nunca assinar consultas a compromissos e criptografar todos os emails armazenados localmente. A última opção é útil porque criptografa as mensagens com sua própria chave – apenas você poderá lê-las no futuro. Se não ativar essa opção, as mensagens criptografadas por você não poderão ser lidas por ninguém – nem você mesmo – depois de enviadas. É também necessário ativar a opção *Confiar sempre nas chaves em meu chaveiro ao criptografar*, caso contrário o Evolution ignorará chaves não assinadas. ➡



**Figura 8:** Use o *Editor de Contas* para especificar as chaves que o *Evolution* deve usar. Em vez do *key ID*, o *Evolution* dá uma colher de chá e permite que seja informado o endereço de email correspondente.

### Quadro 1: Distribuindo e assinando chaves

As comunicações interpessoais protegidas pelo GnuPG requerem que as partes estejam em acordo. Se você quiser enviar uma mensagem criptografada a um amigo, esse seu amigo precisa ter, de antemão, sua chave pública. Não faz lá muito sentido ficar enviando mensagens assinadas a torto e a direito se os destinatários não puderem verificar a autenticidade dessa assinatura e da mensagem. Em ambos os casos, a troca de senhas é um problema sério. Quem garante que a chave que você recebeu por email pertence realmente àquela pessoa? Pode ser que um rufião, usando um email falso, tenha enviado uma chave falsa para você. Nesse caso, se você aceitar essa chave, os emails vindos desse impostor serão considerados como... confiáveis.

Para evitar isso, fazemos uso de *impressões digitais* nas chaves. Essas impressões digitais (ou *fingerprints*) são uma combinação de letras e números que identificam e validam a chave. Você pode gerar uma impressão digital de suas chaves com o comando `gpg --fingerprint key-ID`. Em vez do *key-ID* pode-se informar o endereço de email, desde que cada chave seja exclusiva de um endereço.

Se você receber – por email ou baixado da Internet – uma chave qualquer de criptografia, pode verificar pela impressão digital se essa chave é mesmo de quem parece ser. Essa é uma maneira bastante segura de ter certeza que a chave é autêntica. Para verificar, você pode telefonar ao usuário e perguntar qual a impressão digital dele. Pode ainda reunir-se com ele e trocar impressões digitais em um meio não-conectado – disquetes, por exemplo. Se as impressões digitais baterem, você pode usar o comando `gpg --import arquivo_com_a_chave` e pendurar a chave em seu chaveiro digital.

“Mas e se eu precisar me comunicar com alguém que nunca vi na vida?”, você poderia se perguntar. Realmente, trocar *fingerprints* é prático apenas se o usuário conhecer a pessoa. É aí que entra em cena a figura da *Rede de Confiança* (*Web of Trust*).

Vamos supor que você tenha uma chave em que confia. Você pode, nesse caso, adicionar a *sua* assinatura à chave. É como se você estivesse *endossando*, com sua própria chave (e sua reputação pessoal), a identidade do dono da chave. Se devolver a chave, já assinada por você, para seu dono, ele pode redistribuí-la. Com isso, quem conhecer e confiar em você também vai confiar que aquela chave é autêntica – mesmo que não conheça seu dono. O comando para assinar uma chave é:

```
gpg --sign-key key-ID
```

Para que não seja preciso ficar enviando chaves para lá e para cá por email, existem alguns servidores de chaves na Internet. É possível baixar deles as chaves de milhares de pessoas e organizações. Os servidores formam uma rede sincronizada e, portanto, todos possuem as informações de todas as chaves públicas existentes. A não ser que algum usuário paranóico não envie suas chaves públicas a algum servidor, é provável que qualquer um deles possua todas as chaves de que você precisará em toda a sua vida. O comando:

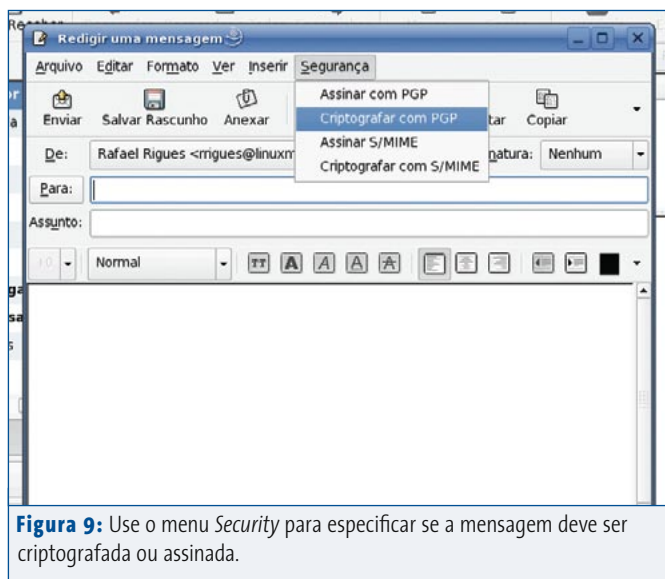
```
gpg --recv-keys key-ID
```

baixa a chave com o identificador indicado (*key ID*) de seu servidor preferido. Já o comando:

```
gpg --send-keys
```

manda suas chaves para o servidor de senhas. Você será instado a confirmar que quer realmente enviar suas chaves, já que o comando envia para o servidor todas as chaves públicas presentes em seu sistema: além das suas chaves públicas, o comando também envia as chaves públicas que você “endossou”. Se quiser atualizar todas as chaves em seu sistema e receber todas as chaves de outros usuários, use o comando:

```
gpg --refresh-keys
```



**Figura 9:** Use o menu *Security* para especificar se a mensagem deve ser criptografada ou assinada.

Ao redigir uma mensagem (**figura 9**), use o menu *Security* para adicionar uma assinatura digital ou cifrar seu conteúdo. Antes de assinar a mensagem, o Evolution solicita a sua frase secreta – afinal, mesmo ele precisa saber se você é realmente o dono daquela chave. Os destinatários ficarão felizes em saber que você usa um cliente de email preocupado a esse ponto com a segurança.

Se for cifrar a mensagem, o Evolution procura no sistema pela chave pública do destinatário. O sistema recusará chaves caso não correspondam ao endereço de email escrito no campo *Para:*, cancelando a ação e mostrando uma mensagem de erro.

O ícone de um cadeado é exibido no rodapé das mensagens criptografadas. Clicar nele leva o usuário a uma caixa de diálogo com detalhes sobre a criptografia e a segurança daquela mensagem em

uma mensagem pela linha de comando.

## Que programa escolher?

Embora todos os programas testados trabalhem com o GnuPG, são bem diferentes entre si. É bem fácil configurar o Evolution para criptografar suas mensagens, o que o torna ideal para usuários não-técnicos que não querem suar para realizar uma tarefa assim simples. Ele é sublime no trato com o moderníssimo padrão OpenPGP/MIME, e desde a versão 2.4 reconhece criptografia *inline*. Se você usa uma versão mais antiga, vale a pena atualizar só por causa desse recurso, senão você terá que salvar em disco as mensagens *inline* e abrir a telinha preta para usar o comando `gpg` – não é o que se pode chamar de moleza...

O Mozilla Thunderbird não suporta o GnuPG por padrão, mas isso é facilmente resolvido com o plugin Enigmail. Já na

instalação, um problema: o usuário tem que digitar o caminho até o utilitário `gpg`, já que o Thunderbird não consegue fazer isso sozinho. As outras configurações são, entretanto, bastante simples. O Thunderbird também poupa dores de cabeça ao reconhecer tanto a criptografia *inline* como o padrão OpenPGP/MIME. O programa também marca alguns gols por seu sistema integrado de administração de chaves locais – mas fica devendo sua contrapartida remota, já que não há lugar algum para administrar servidores de chaves. Além disso, um bug irritante impede o uso do Thunderbird 1.0.6 e a versão mais recente do Enigmail em Português. Veja o **quadro 2** para saber mais.

A maior desvantagem do KMail, ao menos na versão 1.6.2, é a falta de suporte a OpenPGP/MIME; a única maneira de usar OpenPGP/MIME no KMail é instalar o plugin Ägypten, que deve ser compilado a partir do código fonte. Mas, lástima! Em vez disso, faça uma recauchutagem geral e atualize seu KDE 3.2 para uma versão mais nova. Com o KDE 3.3 você leva de presente o KMail 1.7; já o KDE 3.4 dá de brinde o KMail 1.8, cheio de novos recursos como o realce de mensagens assinadas e criptografadas, por exemplo. Os usuários não perderão mais tempo precioso para identificar se a assinatura é ou não válida e confiável, já que o sistema de cores empregado resume tudo numa simples olhadela. ■

### Quadro 2: segurança poliglota

Se você possui uma versão traduzida do Thunderbird (como por exemplo, em Português do Brasil), precisa também de uma versão traduzida do Enigmail. Os “language packs” estão disponíveis na página oficial do programa [6], e são instalados como qualquer outra extensão do Thunderbird. Entretanto, a versão mais recente do Enigmail não funciona com o Thunderbird 1.0.6 em português (uma mensagem de erro em vermelho surge no rodapé da janela do programa). A solução é simples, mas não ideal: desinstale as traduções do Enigmail e do Thunderbird, revertendo-os para o original em inglês, e tudo passa a funcionar corretamente.

### INFORMAÇÕES

- [1] GnuPG: [www.gnupg.org](http://www.gnupg.org)
- [2] KMail: [kmail.kde.org](http://kmail.kde.org)
- [3] Projeto Ägypten: [www.gnupg.org/aegypten](http://www.gnupg.org/aegypten)
- [4] Mozilla Thunderbird: [www.mozilla.org/products/thunderbird](http://www.mozilla.org/products/thunderbird)
- [5] Evolution: [www.gnome.org/projects/evolution](http://www.gnome.org/projects/evolution)
- [6] Enigmail: [enigmail.mozdev.org](http://enigmail.mozdev.org) e [www.thunderbird-mail.de/extensions/enigmail/enigmail.php](http://www.thunderbird-mail.de/extensions/enigmail/enigmail.php)