**Primeiros passos com GnuPG.** **Por:** Leonardo dos Reis Vilela
**Gerando uma nova chave PGP**

**Introdução**

O PGP (abreviação de Pretty Good Privacy, ou Muito Boa Privacidade) é um programa de criptografia de chave pública altamente seguro, originalmente escrito por Philip Zimmermann. Nos últimos anos o PGP conquistou milhares de entusiastas em todo o mundo e tornou-se de fato um padrão para a criptografia de correio eletrônico.

Segue-se abaixo um tutorial rápido para criar uma chave pública e uma chave privada. O tutorial foi baseado no gnupg (gpg). Usuários GNU/Linux e usuários MS/Windows poderão seguir este tutorial, pois a sintaxe (os comandos) são relativamente os mesmos.

Caso não tenha instalado o "gnupg", utilize os links a seguir para fazer o download :

GnuPG 1.2.2 - Dos/Windows
Basta descompactar o arquivo em alguma pasta (ex. c:\gnupg). O "gnupg" que estamos à instalar é somente para a console, ou seja, mesmo no DOS ou Windows teremos que digitar as linhas de comandos. Depois de criar a pasta e descompactar o arquivo copiado, entre no *Prompt do MS-DOS* e faça "*cd \gnupg*" (se utilizaste o nome "gnupg" para a pasta).
Agora podes seguir o tutorial, criar as chaves, cifrar etc.

GnuPG 1.2.2 - GNU/Linux
Deixamos os comentários sobre a instalação do "gnupg" sobre o GNU/Linux de lado, pois praticamente todas as distribuições o trazem instalado.

**Entenda os termos mais utilizados.**

**Chaves** : Uma chave não é nada mais do que um valor, uma sequência de digitos, uma cadeia de valores etc que será utilizada por um algoritmo para cifrar ou decifrar dados. Aqui no "gnupg" as chaves serão armazenadas em arquivos.

**Chave pública** : Esta chave é utilizada para cifrar dados. Um algoritmo utilizará esta chave para encriptar os dados, estes dados depois de cifrados podem ser lidos apenas com a chave privada que completa o par com essa chave pública. Chamamos de chave pública pois é uma chave que pode ser liberada sem nenhum critério, podemos distribuir nossa chave pública a todos, e este é o objetivo. Para que encriptem os dados que nos serão enviados, as pessoas precisarão de nossa chave pública.

**Chave privada** : Esta chave é utilizada para decifrar dados. O mesmo algoritmo que utilizamos para cifrar agora será utilizado para decifrar, mas utilizamos agora uma chave privada. Esta chave deve ser guardada "*a sete chaves*", esta chave não deve e nem pode ser distribuida, pois ela é destinada somente ao utilizador. A chave privada tem dois objetivos principais : o primeiro é de decifrar os dados que foram anteriormente cifrados com a nossa chave pública e o segundo é de assinar (fazer assinaturas digitais). O objetivo de assinar algum documento ou email é de dizer que este é verdadeiro e que garantimos a autenticidade do mesmo. Veja um exemplo : Josafah tem que enviar um documento ao seu Banco, mas como o Banco pode saber se foi ele mesmo que enviou o documento ? Então Josafah pode assinar o email com a sua chave privada e isto quer dizer : "Quem está enviando o email sou eu mesmo".

**Frase secreta / Senha** : Toda vez que for preciso acessar a chave privada (decifrar dados ou assinar), então esta senha será solicitada.

Observação : cada utilizador tem um par de chaves (uma privada e uma pública).

## Gerando as chaves públicas e privadas.

## Passo 1. Início com o GPG :

O gnupg (gpg) - durante a primeira vez que for executado com uma certa conta (login) - irá criar um diretório e um arquivo de opções para este usuário.

$ *gpg --gen-key*

gpg (GnuPG) 1.0.7; Copyright (C) 2002 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.
gpg: Aviso: usando memória insegura!
gpg: por favor veja http://www.gnupg.org/faq.html para mais informações
gpg: /home/user1/.gnupg: diretório criado
gpg: /home/user1/.gnupg/options: novo arquivo de opções criado
gpg: você deve reiniciar o GnuPG, para que ele possa ler o novo arquivo de opções
Se não teve nenhuma mensagem de erro, então continuemos ao passo 2.

Neste primeiro passo não fizemos que criar o diretório ".gnupg" e o arquivo "options".
Observação : dependendo da versão do gnupg ele criará os arquivos e já iniciará a criação do par de chaves, caso ele faça este, vá para o "passo 2" deste tutorial.

**Passo 2. Criando uma chave privada.**

Iremos executar novamente o gpg, e então ele criará um par de chaves (uma chave pública e uma chave privada). Será perguntado algo como o tipo de chave, o tamanho da chave, uma frase para a senha etc. Basta responder de acordo com o que será pedido, aqui em baixo temos um exemplo criado com o nome de utilizador 'Baby Sauro'.

Para continuar, basta executar "*gpg -gen-key*".

$ *gpg --gen-key*

gpg: WARNING: --no-auto-key-retrieve is a deprecated option.
gpg: please use "--keyserver-options no-auto-key-retrieve" instead
gpg (GnuPG) 1.0.7; Copyright (C) 2002 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.
gpg: Aviso: usando memória insegura!
gpg: por favor veja http://www.gnupg.org/faq.html para mais informações
gpg: porta-chaves `/home/user1/.gnupg/secring.gpg' criado
gpg: porta-chaves `/home/user1/.gnupg/pubring.gpg' criado
Por favor selecione o tipo de chave desejado:
**(1) DSA e ElGamal (padrão)**
(2) DSA (apenas assinatura)
(4) ElGamal (assinatura e criptografia)
(5) RSA (apenas assinatura)

Sua opção? **1**

O par de chaves DSA terá 1024 bits.
Prestes a gerar novo par de chaves ELG-E.
tamanho mínimo é 768 bits
tamanho padrão é 1024 bits
tamanho máximo sugerido é 2048 bits
Que tamanho de chave você quer? (1024) **2048**

//Aqui escolhemos 2048 bits, mas podemos utilizar 4096 etc.

O tamanho de chave pedido é 2048 bits

Por favor especifique por quanto tempo a chave deve ser válida.
**0 = chave não expira**
<n> = chave expira em n dias
<n>w = chave expira em n semanas
<n>m = chave expira em n meses
<n>y = chave expira em n anos

A chave é valida por? (0) **0**

A Key não expira nunca
Está correto (s/n)? **s**

Você precisa de um identificador de usuário para identificar sua chave; o programa constrói o identificador a partir do Nome Completo, Comentário e Endereço Eletrônico desta forma:

"Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Nome completo: **Baby Sauro**
Endereço de correio eletrônico: **saurojunior@silva.marte**
Comentário:
Você selecionou este identificador de usuário:
"Baby Sauro <saurojunior@silva.marte>"
Muda (N)ome, (C)omentário, (E)ndereço ou (O)k/(S)air? **O**

Você precisa de uma frase secreta para proteger sua chave. **Afrasepoderahserinteligente**

gpg: /home/user1/.gnupg/trustdb.gpg: banco de dados de confiabilidade criado chaves pública e privada criadas e assinadas.
Chave marcada como de confiança absoluta

pub 1024D/**FB55C0C4** 2004-03-09 Baby Sauro saurojunior@silva.marte
Impressão da chave = **3191 C8A9 FEE9 8F89 E955 8616 E00C 0A4E FB55 C0C4**
sub 2048g/D0E62952 2004-03-09

A Impressão da Chave é o fingerprint. (No final do documento mostraremos como recuperar o ID da chave, fingerprint etc). Observação : veja que o **IDENTIFICADOR** da chave é o final do fingerprint. Neste caso temos o ID = **FB55C0C4** e o final do fingerprint: ... E00C 0A4E **FB55 C0C4**

Recomendamos guardar o ID e o fingerprint (escreva em algum papel) - somente para facilitar o aprendizado (pois existem formas de recuperá-lo depois).

Vamos testar ? Que tal encriptar um documento e depois tentar lê-lo?

Para cifrar qualquer arquivo, basta fazer :

$ gpg -f ***Nome-do-Arquivo***
Será solicitado o ID da chave. Pode ser digitado tanto o ID da chave, quanto o nome do utilizador, o email etc. Em nosso exemplo iremos digitar o nosso nome "Baby Sauro" que o gnupg encontrará automaticamente a chave que procuramos (que na verdade é a nossa chave). Neste exemplo estamos cifrando um documento com a nossa chave pública, podemos cifrar um documento com a chave pública de outra pessoa, mas para frente explicaremos os métodos para importar chaves.

Se nenhum erro, um arquivo "Nome-do-Arquivo.gpg" foi criado. Este arquivo é o arquivo cifrado

através de nossa chave pública. Podemos agora transportar o documento (através de email, discos, memorias flashs etc) sem nenhuma preocupação - pois para o ler é preciso da chave privada que completa o par com a chave pública que foi utilizada (que no caso é a nossa). Devemos relembrar que para acessar a chave privada, é preciso utilizar a *frase secreta / senha.* Caso queiras, pode até mesmo apagar o documento original, quando precisar ler o documento, basta decrifar o "Nome-do-Arquivo.gpg".

Apenas para fazer um teste, vamos decifrar o documento. Faça :

$ gpg -d **Nome-do-Arquivo.gpg**
Será solicitado a frase secreta.

**Passo 3. Importando uma chave pública (de outro utilizador)**

Objetivo: Imagine que queira enviar algum documento sigiloso para alguém ou algo deste tipo. A maneira mais simples será de utilizar a chave pública da pessoa e criptografar(cifrar) o documento. Depois disto feito, o documento somente poderá ser lido pelo utilizador proprietário da chave pública (pois ela tem a chave privada que completa o par da chave pública).

Para obter uma chave pública de alguém existem diversas maneiras, para simplificar o processo, podemos entrar diretamente em algum site de *server pgp* e fazer a cópia da chave, para isto podemos utilizar http://pgp.mit.edu , http://www.keyserver.net etc.

Observação : A porta utilizada pelo PGP é a ***11371/tcp***, mas talvez o administrador de tua rede tenha bloqueado ou o firewall não permita. Para eliminar estes tipos de "blocagem" que não permita seguir este tutorial, iremos utilizar um servidor que permite extrair chaves diretamente da porta 80 (este servidor ainda é experimental).

Entremos em :

http://keyserver.kjsl.com/~jharris/skylane/pks-commands.php#extract

Escolha "Extract a key from the server".
Coloque o nome do utilizador de quem queres copiar a chave.
Como exemplo iremos copiar a chave de "Leonardo dos Reis Vilela".

Podemos escolher "Index" e "Verbose Index". Este permite visualizar as assinaturas do utilizador e esse permite visualizar somente o fingerprint e o ID do utilizador. Iremos marcar "Index".

Dentro da caixa de texto "Search String" coloque "Leonardo dos Reis Vilela" e acesse "Do the search!".

Pronto. Se tudo ocorreu bem, temos algo como :

pub 1024D/D75FE7B1 2004-01-25 Leonardo dos Reis Vilela <leo@mundociencia.com.br>
Key fingerprint = 24FB 5FC1 6520 C975 7F5D F98F E7E1 CFE0 D75F E7B1

Clique sobre o ID da chave (clique sobre D75FE7B1).

Teremos na tela a chave de "Leonardo dos Reis Vilela". Faça um "selecionar tudo" e grave tudo dentro de um arquivo somente texto (utilize o vi, notepad etc Nada de utilizar formatos como .DOC, .XLS, OpenOffice...). Gravemos o arquivo com o nome de "PublicaLeonardoReis.asc" e para facilitar coloque o arquivo no mesmo caminho que temos o gnupg (c:\gnupg como no nosso exemplo).

faça agora :

$gpg --import PublicaLeonardoReis.asc

Veja o resultado feito no MS/Windows XP :

C:\gnupg>gpg --import PublicaLeonardoReis.asc
gpg: key D75FE7B1: public key "Leonardo dos Reis Vilela <leo@mundociencia.com.br
>" imported
gpg: Total number processed: 1
gpg: imported: 1

Podemos ver aqui que o ID da chave importada é : "0xD75FE7B1"

No GNU/Linux para importar diretamente uma chave de um servidor através da porta 80 (sabendo o ID da chave), bastaria fazer :

$ *wget 'http://keyserver.kjsl.com/~jharris/skylane/pks-commands.php?docmd=lookup&op=get&search=0xD75FE7B1' -O LeonardoReis.asc*

E depois importamos a chave :

[user1@sol]$ *gpg --import LeonardoReis.asc*

Neste momento já temos a chave pública de "*Leonardo dos Reis Vilela*", caso queiras enviar um e-mail para ele ou cifrar um documento com a sua chave (com a chave de Leonardo dos Reis Vilela), então veja o exemplo abaixo :

Primeiro nós criaremos um arquivo de teste (que será cifrado).

O arquivo que queremos cifrar é (por exemplo) o "textopuro". Agora cifrando-o :

$ *gpg -f textopuro*

Siga os passos ele pedirá o identificador do utilizador (coloque o ID da chave ou então o próprio nome do utilizador). Faça a confirmação (normalmente pressione o "enter") ...

Um arquivo chamado "extopuro.gpg" será criado. Este arquivo agora está cifrado com a chave pública do utilizador que foi escolhido, somente quem tem a chave privada desta chave pública poderá ler este arquivo agora.

Voltando...

Observação: uma outra maneira de obter uma chave seria importá-la diretamente do servidor (para isto ele fará uma conexão na porta 11371, assegure que o teu firewall permita esta conexão). Este comando abaixo permite elimintar toda a burocracia de entrar em sites e copiar a chave.

*$ gpg --keyserver pgp.mit.edu --keyserver-option no-auto-key-retrieve --recv-keys 0xXXXXXXXX --import 0xXXXXXXXX*

Importando a chave de "Leonardo dos Reis Vilela" diretamente através do gpg :

$ *gpg --keyserver pgp.mit.edu --keyserver-option no-auto-key-retrieve --recv-keys 0xD75FE7B1 --import 0xD75FE7B1*

Tudo isto, onde :
--keyserver <servidor>, exemplos : pgp.mit.edu, www.keyserver.net etc.
--recv-keys <ID da chave>
--import <ID da chave>

**Passo 4. Assinando uma chave pública**

Por que assinar chaves ? Alguém quer te enviar um documento que contem dados sigilosos, então ele pedirá a tua chave pública (ou então fará o download em algum site como: pgp.mit.edu etc). Ele faz o download ou recebe a chave diretamente de ti por email. Mas agora surge o grande problema: como saber se a chave que ele tem em mãos é realmente a *tua chave*? Outra pessoa pode ter criado uma chave com o teu nome por exemplo, e tê-la enviado no teu lugar. Mas uma forma de *minimizar* a desconfiança é basear nas assinaturas de tua chave. Tomamos como exemplo que ele conheça o Joãozinho (e o Joãozinho assinou a tua chave), então se ele conhece o Joãozinho e o Joãozinho assinou a tua chave, então ele pode fazer confiança em tua chave, ele sabe agora realmente que a chave de quem ele fez o download é a tua (pois ele confia em Joãozinho e Joãozinho confia em ti, então vamos pela transitividade e acabamos de criar uma cadeia de confiança).

Observação: Os passos 4 e 5 talvez não serão executados por ti neste momento (pois talvez não tenha nenhuma chave para assinar neste momento), mas leia, é preciso saber que estes conceitos existem.

O ato de assinar uma chave é o ato de reconhecimento que a chave que vai ser assinada é realmente da pessoa (que se diz dona da chave).

Para assinar uma chave, é um dever observar o seguinte:

**4.1**. O proprietário da chave convence o assinador (tu no caso) que a identidade na chave é verdadeiramente sua própria identidade (comparar se é o mesmo nome). Isto está à dizer que o proprietário da chave deve apresentar uma identificação expedida pelo governo (policia federal etc) com uma foto e informação que coincida com o proprietário da chave. Geralmente o documento utilizado é o passaporte (pois é internacional).

**4.2**.Depois disto é preciso verificar se o *fingerprint* que o proprietário mostra é o mesmo fingerprint da chave que vai ser assinada.

Claro que quando o Joãozinho foi assinar a tua chave, é necessário que ele tenha verificado com cuidado que a chave pertencia realmente à ti, senão a cadeia de confianças quebra.

Poedmos dizer que o mais importante não é ter muitas assinaturas, mas sim, assinaturas confiáveis.

Faça de conta que verificamos o *passaporte* e o *fingerprint* de Leonardo dos Reis Vilela e tudo coincidiu com as informações de sua chave. Então poderemos assinar a sua chave (a sua chave = a chave de Leonardo dos Reis Vilela).

Iremos assinar a chave de "Leonardo dos Reis Vilela". Como já conhecemos o ID da chave de Leonardo, basta fazer: (observação : para assinar uma chave, é necessário ter importado a chave publica que será assinada).

$ *gpg --edit-key D75FE7B1*
Observação: caso não goste de hexadecimais, coloque o nome do utilizador. :-)
$ *gpg --edit-key Leonardo dos Reis Vilela*

gpg: WARNING: --no-auto-key-retrieve is a deprecated option.
gpg: please use "--keyserver-options no-auto-key-retrieve" instead
gpg (GnuPG) 1.0.7; Copyright (C) 2002 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.
gpg: Aviso: usando memória insegura!

gpg: por favor veja http://www.gnupg.org/faq.html para mais informações
gpg: a verificar a base de dados de confiança
gpg: a verificar à profundidade 0 assinado=0 ot(-/q/n/m/f/u)=0/0/0/0/0/1

*pub 1024D/D75FE7B1 criada: 2004-01-25 expira: never confiança: -/-*
sub 4096g/7101E62D criada: 2004-01-25 expira: never

(1). **Leonardo dos Reis Vilela** <leo@mundociencia.com.br>
Comando> **sign**
pub 1024D/D75FE7B1 criada: 2004-01-25 expira: never confiança: -/-
Impressão digital: 24FB 5FC1 6520 C975 7F5D F98F E7E1 CFE0 D75F E7B1

//Verifique com atenção se o *fingerprint* aqui é o mesmo que ele tinha te passado.
//O ID são os 8 ultimos hexadecimais do *fingerprint.*

Leonardo dos Reis Vilela <leo@mundociencia.com.br>

Com que cuidado é que verificou que chave que está prestes a assinar pertence à pessoa correcta? Se não sabe o que responder, escolha "0".
(0) Não vou responder. (default)
(1) Não verifiquei.
(2) Verifiquei por alto.
(3) Verifiquei com bastante cuidado.

Sua opção? **2** (responda de acordo com o tenha feito...)

Você tem certeza de que quer assinar esta chave com
sua chave: "Baby Sauro <saurojunior@silva.marte>"

Verifiquei esta chave com muito cuidado.

//Se tudo vai bem, assine.
Realmente assinar? S
…
Command> **quit**
Save changes? **Yes**

Atenção: é preciso verificar o passaporte/documento de identidade válido para assinar uma chave (pois senão, como irá realmente saber se a chave que está assinando é realmente da pessoa que quer assinar? Até mesmo pode existir duas pessoas com o mesmo nome, então é necessário verificar o passaporte - ou o documento de identidade no mínimo - da pessoa e que ela mostre o *fingerprint* de sua chave, pois se existem duas pessoas com o mesmo nome, certamente o *fingerprint* será diferente para cada chave, e é importante que o proprietário diga qual é o seu *fingerprint*).

**Passo 5. Criando um arquivo com a chave pública assinada.**

Depois de ter assinado a chave de alguém é necessário enviá-la ao seu proprietário ou então submetê-la a um servidor PGP. Por exemplo: pgp.mit.edu , www.keyserver.net … Se não for feito nenhum destes dois passos, a assinatura não terá nenhum efeito. O normal de se fazer é realizar os dois (submeter a chave à algum servidor e depois enviá-la ao seu proprietário).

Nota do autor : Caso tenha feito o passo 4 e tenha assinado a minha chave, não publique a chave em nenhum site (pois tu não sabes se eu sou realmente que eu digo ser, precisa ver o meu passaporte e que eu te mostre o meu fingerprint).

Caso queira que o próprio proprietário da chave faça a submissão, então envie somente o arquivo (com a chave assinada) para o proprietário. Para fazer isto, basta exportar a chave.

$ *gpg -a --export D75FE7B1 > LeonardoReisSigned.asc*

gpg: Warning: using insecure memory!

Agora envie o arquivo 'LeonardoReisSigned.asc' para o seu email (para o email do proprietário da chave, no caso seria leo@mundociencia.com.br).

**Passo 6. Extraindo a minha própria chave pública.**

Caso alguém queira te enviar um documento ou um email cifrado com a tua chave pública, é necessário que a pessoa tenha a tua chave pública. Existem algumas maneiras da pessoa conseguir a chave: seja que ela faça uma cópia da chave de algum servidor pgp (pgp.mit.edu por exemplo), seja que ela faça um pedido diretamente ao proprietário da chave etc.

Partindo do ponto que a pessoa te fez um pedido da tua chave pública, então é necessário criar um arquivo com a tua chave e passar o arquivo para o pedinte (por exemplo, podemos passar pelo email) :

Neste exemplo baseamos que o ID de nossa chave é o "*FB55C0C4*":

$ *gpg -a --export FB55C0C4 > MinhaChave.asc*

gpg: WARNING: --no-auto-key-retrieve is a deprecated option.
gpg: please use "--keyserver-options no-auto-key-retrieve" instead
gpg: Aviso: usando memória insegura!
gpg: por favor veja http://www.gnupg.org/faq.html para mais informações

Onde "*FB55C0C4*" é o ID da chave (da chave que criamos aqui no exemplo, coloque aqui o teu ID) e " MinhaChave.asc " é onde será gravada a chave.

Agora basta que envie a chave pública para a pessoa e então ela poderá criptografar o email ou algum documento com a tua chave pública. Se foi criptografado com a tua chave pública, somente a tua chave privada será capaz de ler o documento (e a frase secreta de tua chave será requisitada).

**7. Outros comandos importantes:**

**7.1)Para ver a tua lista de assinaturas (quem assinou a tua chave):**
$ *gpg --list-sigs*

**7.2)Para ver a lista de chaves (as chaves publicas que tu tens):**
$ *gpg --list-keys*

**7.3)Para ver o IDENTIFICADOR de tua chave e também o fingerprint completo :**

$ *gpg --fingerprint "Baby Sauro"* //onde "Baby Sauro" será substituído por teu nome.

gpg: WARNING: --no-auto-key-retrieve is a deprecated option.
gpg: please use "--keyserver-options no-auto-key-retrieve" instead
gpg: Aviso: usando memória insegura!
gpg: por favor veja http://www.gnupg.org/faq.html para mais informações
pub 1024D/FB55C0C4 2004-03-09 Baby Sauro saurojunior@silva.marte
Impressão da chave = 3191 C8A9 FEE9 8F89 E955 8616 E00C 0A4E FB55 C0C4
sub 2048g/D0E62952 2004-03-09

**7.4)** Agora é preciso disponibilizar a tua chave pública em algum servidor de PGP (para que as pessoas que precisem de tua chave pública, a achem). Para isto entre em http://pgp.mit.edu ou em outro servidor como por exemplo : http://www.keyserver.net . Não importa em qual servidor, pois após a submissão, o servidor sincronizará com o restante dos servidores, e todos os

servidores terão a tua chave.

**7.5)** Caso alguém assine a tua chave e submeta a tua chave sobre algum servidor PGP, para recuperar a assinatura que a pessoa fez em tua chave e para atualizar a tua chave (pois uma nova pessoa assinou a tua chave), faça:

*$ gpg --keyserver pgp.mit.edu --keyserver-option no-auto-key-retrieve --recv-keys 0xXXXXXXXX --import 0xXXXXXXXX*

Onde 0xXXXXXXXX é o ID de tua chave.

Neste exemplo, estamos utilizando o servidor pgp.mit.edu. Onde XXXXXXXX é o ID de tua chave.

**7.6)** Um pequeno exercício, crie um arquivo e o cifre com a minha chave pública (Leonardo dos Reis Vilela) e envie para o meu e-mail (leo@mundociencia.com.br). Irei te responder com outro e-mail cifrado, será preciso que utilize a tua chave privada para ler o documento. Para que eu possa te responder, eu irei procurar a tua chave em algum servidor PGP, então é preciso que faça a submissão de tua chave em algum servidor PGP (veja 7.4 para maiores informações).

Dicas a seguir:

**7.6.1** Copie a minha chave (de http://pgp.mit.edu por exemplo, ou outro servidor.).
**7.6.2** Crie um documento e o cifre (*gpg -f nomedoarquivo*)
**7.6.3** Envie o documento cifrado para o meu e-mail.

**7.7) Decifrando um documento.**

$ *gpg -d arquivo.*
Alguns dados serão perguntados (frase/senha para acessar a chave privada etc).

**7.8)Para ver o ID e também o fingerprint de uma chave (de outro utilizador):**

$ *gpg --fingerprint "Nome ..."*
Onde "Nome ..." será substituído pelo nome que se procura. Obs.: Isto irá fazer uma busca entre as chaves publicas que já foram por ti importadas. Caso ela não tenha sido importada e queiras saber o fingerprint de alguém, basta entrar em algum servidor de pgp.

---

Caso encontre algum erro ou queira fazer algum acréscimo, avise:
pub 1024D/D75FE7B1 2004-01-25 Leonardo dos Reis Vilela leo@mundociencia.com.br
1. Impressão da chave = 24FB 5FC1 6520 C975 7F5D F98F E7E1 CFE0 D75F E7B1
2.sub 4096g/7101E62D 2004-01-25

Leonardo dos Reis Vilela
leo@mundociencia.com.br