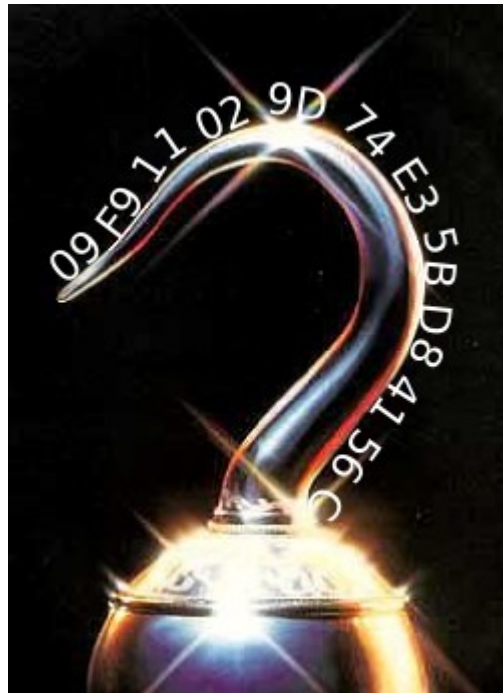


Credson Isaac L dos Santos

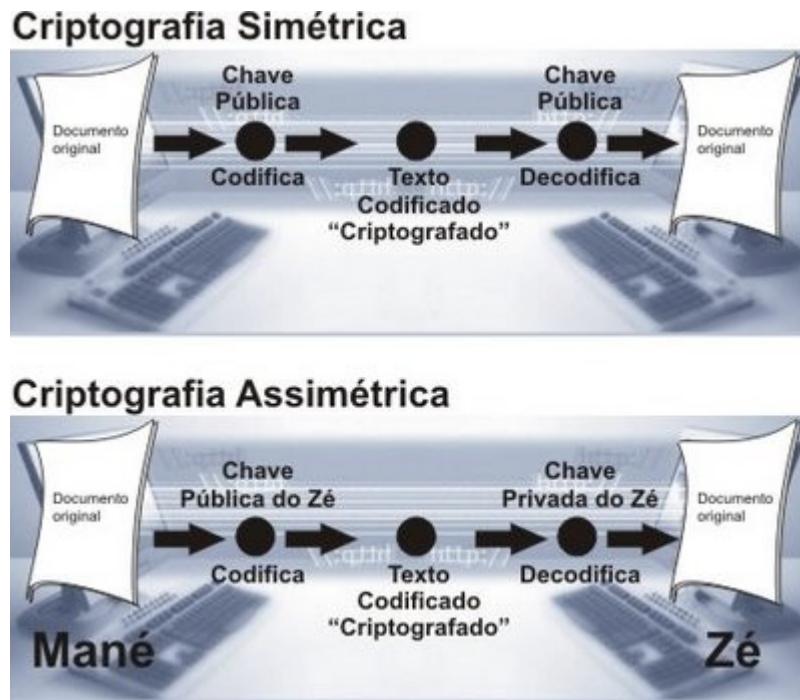
# TUTORIAL

## Entendendo o gpg (GnuPG)



O GPG (GnuPG) usa o sistema de chaves pública e privada (Assimétrico) para assegurar a transferência de informações entre usuarios remotos. É crucial neste método que a chave privada seja guardada pelo criador e somente por este usada. Esta chave não deve ser trafegada pela Internet.

Veja na ilustração a diferença entre o sistema simétrico para o assimétrico:



A instalação do GPG (GnuPG) é bastante simples, na maioria das distribuições atuais já possuem este pacote instalado,. Nos sistemas baseados em Debian, podemos instala-los via apt-get.

Este tutorial tem a finalidade de mostrar alguns comandos do GPG exemplificando na prática os conceitos de criptografia e envio de mensagem bem como sua recepção e descryptografia.

Vamos praticar:

### **Passo 1 Criação de usuario e senha:**

```
root@isaac-laptop:/# useradd -m -c "credson isaac" -g "casa" \bin\bash isaac
```

```
root@isaac-laptop:/# useradd -m -c "maria alice" -g "casa" \bin\bash alice
```

### **Passo 2 Criação do par de chaves utilizando a ferramenta gpg (GNUPG):**

Iremos executar 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 'credson isaac'.

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

```
root@credson-laptop:/home/credson# gpg --gen-key  
gpg (GnuPG) 1.4.9; Copyright (C) 2008 Free Software Foundation, Inc.  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.
```

Por favor selecione o tipo de chave desejado:

- (1) DSA e Elgamal (padrão)
- (2) DSA (apenas assinatura)
- (5) RSA (apenas assinar)

Sua opção? **1**

par de chaves DSA vai ter 1024 bits.

ELG-E chaves podem ter o seu comprimento entre 1024 e 4096 bits.

Que tamanho de chave você quer? (2048) **2000**

O tamanho de chave pedido é 2000 bits

arredondado para 2016 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 chave 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: **bob esponja**

Endereço de correio eletrônico: **bob@esponja.com.br**

Comentário: **minha chave aleatoria é esta, isso é confidencial**

Você está usando o conjunto de caracteres `utf-8'.

Você selecionou este identificador de usuário:

```
"bob esponja (minha chave aleatoria é esta, isso é confidencial) <bob@esponja.com.br>"
```

Muda (N)ome, (C)omentário, (E)ndereço ou (O)k/(S)air? **o**

Você precisa de uma frase secreta para proteger sua chave.

gpg: gpg-agent não está disponível nesta sessão

A frase secreta não foi repetida corretamente; tente outra vez.

Precisamos gerar muitos bytes aleatórios. É uma boa idéia realizar outra atividade (digitar no teclado, mover o mouse, usar os discos) durante a geração dos números primos; isso dá ao gerador de números aleatórios uma chance melhor de conseguir entropia suficiente.

```
+++++++.....>+++++.....+++++
+++++.+++++.+++++..+++++.....+++++.....+++++
+++++.....>+++++.....+++++
```

Precisamos gerar muitos bytes aleatórios. É uma boa idéia realizar outra atividade (digitar no teclado, mover o mouse, usar os discos) durante a geração dos números primos; isso dá ao gerador de números aleatórios uma chance melhor de conseguir entropia suficiente.

```
+++++.....+++++.....+++++.....+++++.....+++++.....
+++++.....+++++.....+++++.....+++++.....+++++.....
+++++.....>+++++>+++++>+++++.....<..++
+++.....>+++++.....<+++++...>..+++++.....<+++++.....+++++
+^^^
```

gpg: chave **1A09B867** marcada como plenamente confiável  
chaves pública e privada criadas e assinadas.

gpg: checando o trustdb

gpg: 3 parcial(is) necessária(s), 1 completa(s) necessária(s), modelo de confiança PGP

gpg: profundidade: 0 válidas: 2 assinadas: 0 confiança: 0-, 0q, 0n, 0m, 0f, 2u

pub 1024D/1A09B867 2009-10-18

Impressão digital da chave: 512E 4EDF 8A9D 9C4C 4414 BAAD **7DF4 BF7C 1A09 B867**

uid bob esponja (minha chave aleatoria é esta, isso é confidencial)

<bob@esponja.com.br>

sub 2016g/7BFC142D 2009-10-18

Então como o shell mesmo descreve, foi gerado sua chave.

Observação : veja que o **IDENTIFICADOR** da chave é o final do fingerprint. Neste caso temos o ID = **1A09B867** e o final do fingerprint: ... **7DF4 BF7C 1A09 B867**

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

### Passo 3 Exportando e importando uma chave.

```
gpg --export -a alice >alice_public.key
```

```
root@credson-laptop:/home/credson# gpg --export -a bob >bob_public.key
```

```
root@credson-laptop:/home/credson# cat bob_public.key
```

A chave pode ser visualizada no arquivo “bob\_public.key” localizado no /home/usuario ativo.

```

root@isaac-laptop: /home/isaac/.gnupg
GNU nano 2.0.9                               Arquivo: isaac pub-key.key
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.9 (GNU/Linux)

m0G1BERKCFcRBACvFWwRB2fJLf39aUA9pSzkjrv80jBrH+4EFmerxApkMx2drKxt
zvHavLHLGmk++b0zqhMoSndRwHbKs1kf3UrvytJ11VqjYkXhLv0c3hc698B7TSI
T5azwPyL046g/h5f9eYU0V38Begt70xbYHwL9krhityL5CLRxbgXrsR3AWcm9xb
EccA3LCpyCTCCyML+D1M7Pcd/jcSHTkpk/ch99ZgnkXbl1X4WJxdagh41rDrTuqT
PAJ7Lue3UyaSked3X55VNjMYn/7cAoYJ3zE1MKntXrBy/PPdYMLhLVAUJntwbl1
YaHhE608g/fXbj7orvbw4hBeLQYD0P4Z9d6407JKCK6LWnFmentFhFuHELwzhdB+
T2zWA/9kxhuQH1rjpcZb6gCYIZF002Msor+I27/fPtXl+LhTxyWoNld/Dw9keSPk
4eDCoPm5Vr7cA8F1+o1bXt3rr9epNPjNLT1GpwTtkLTIWouWPsejqt/H8JmGdSt
8Hm4kt898EgEAToImHVX224CI2Yd+0fDwZ1z5SA9pY175dcbQqaXNhyMgKHLh
aG9vKSA8Y3JLZHNvbl9pc2FhY0B5YWhvby5jb20uYnI+iGYEEcACyFAkrKCFcC
GwMFCQEHrAAGCwKIBwMCCBUCAMEFgIDA0IeAQIXgAAKCRD7Wp5BggGDe5LrAJ9F
82J1i05AwJ0ncUjPwU3ApuYiuACfUJwE2iK1U81c2V704ofP2z7F5S55AgUESSoI
VxAH4L3h7zN5LMA5IL0g5HfVjz/qqL8DRa/xL3MaIrXT06G1dD76+MSD6v+07v8
U2ZS19WPzhdNDaudMprCwi9+1R+LzjHka5KqHnbY055KI9fa8wmIDktsrHd6bBXE
kntpuJaUBEPg1AxfKwImIwqMC0gIhBUcmGVGpEHDrtZmaHyTbSbz5itgvV88d8R5
TAbzueYnck+Hpa4wXck0mZncbhWRW5k8rkYns8svrEIL4pKRw+zCUbZ89nff7YImz
vySEBL9nVLTf+TDSwUdcxUjYh70zUjwzhH0JWVRJE7hZqYVzLcqRBT+TVpAtYChj
bHESTzumYdVHITRUFUsY7wAD80fdGx/8g2NE91aYtc0qgL2AKT2p3ai+u58Mgom7
sKI0t+20Z4bURSDtySjGpp6Ppu4tKNEjEk0Sm21kqTv8gV5muDEGVWbTphcF1j
j1k1JdW3Gue4JgZ1dp8KhpjMCFcasA7APNUm8We4zncdEKHzS3sSwRIyv51cP
66bDTA09vuUNEFs/nXwRldVyJMLWR1LEcYcFr5Tbkk1pMGWQLr7qBeYwRwLM60
+ZJAUrVE07tAmrGLx3+ACLzxbokInx+Pa05Ygx6abpTZ1gDd1jSzKmg2L9cs8Bd
tLbwdT0tccsFNBM7AKSG3IopIp9tB9q5CLAA0+ae1ac1E8EBECA8FAkrKCFcC
GwMFCQEHrAAACgkQ+1qe0R0Bg3v8YQCfRC/0ISVEHDLZLs+D/wiaCL0GD6cAnAyG
0S14Ax4FUY1BW5p9ItJPoZWQm0GiBERKQeMRBAC+Dkjd1nUT3B44ueb0Ci fmfwm
j0rDt2bMBMfctIDqzKi60Z9qu03JRJH4LnYx/yLFvZsvSI2yGj4P9snBQ8rzS96n
FR2RDHjZxc2YwSKpMdh4W/KMPPnUK8Pj+/FbFDRF+L1vP0dJ8b/JY3+9fdkPkg
Bj4JL85C152n0CIumwCguXhX+xMxChEz/t8Z7wM0Zkv/qXMD/RC1QnL0oR+D+hv
Bn1M0tu09pVs/fAZaDcrBIyH7DmCH+LoMjLk8KMfndg15j1s9fC0cGpFa8Bw+onw
SpGXuefCaChK4JL1AJHfpm8nYMLcxcz5a6bPVXwvRca2ASZwmGeae0vzuvrI
Xm6/DUYyCpHRfshRhtc6z08NtApA/9pehKaBlc3Xp9tylDI7LGC18aCa/3m4eI7
rntZguZStj0g1Y1Xo4eS0BVH4LIPAPC+XkbtXZ5r69/Y0gFBarIKNyBAH0J1i6U
+X3azhTf3n5XzCaTES2axtvnTLM/mwXOX+dut64A0QC+fncBlTKv0XiRpsNmIdZ

```

Veja a chave na integra pós comando “gpg –export ”

Exportando para um servidor on-line:

Neste caso usaremos o [pgp.mit.edu](http://pgp.mit.edu) , porém há varios servidores on-line, e quando postado em apenas um deles, eles se atualizarão entra si, e logo logo todos terão sua chave disponivel.

```
root@bob-laptop:/home/bob# gpg --keyserver pgp.mit.edu --send-key 1A09B867
```

O comando para pesquisar uma chave no servidor é:

```
gpg --keyserver keyserver.bu.edu --search-key torvalds
```

O comando para receber uma chave no servidor é:

```
gpg --keyserver keyserver.bu.edu --recv-key torvalds
```

Finalmente, pra não precisar ficar escrevendo "--keyserver keyserver.bu.edu" o tempo todo, você pode colocar, no seu ~/.gnupg/gpg.conf (criando-o, se ele não existir),

```
keyserver keyserver.bu.edu
```

E pronto, agora basta fazer "gpg --send-key 0x00000000" etc.

### Passo 3 Assinando a chave publica.

O comando para assinatura é “sign” no “--edit-key”

```
root@bob-laptop:/home/bob# gpg --edit-key 1A09B867
```

```
gpg (GnuPG) 1.4.9; Copyright (C) 2008 Free Software Foundation, Inc.
```

```
This is free software: you are free to change and redistribute it.
```

There is NO WARRANTY, to the extent permitted by law.

Chave secreta disponível.

```
pub 1024D/1A09B867 criado: 2009-10-18 expira: nunca    uso: SC
      confiança: plena    validade: plena
sub 2016g/7BFC142D criado: 2009-10-18 expira: nunca    uso: E
[plena] (1). bob esponja (minha chave aleatoria é esta, isso é confidencial) <bob@esponja.com.br>
Comando> sign
```

```
pub 1024D/1A09B867 criado: 2009-10-18 expira: nunca    uso: SC
      confiança: plena    validade: plena
Impressão digital da chave primária: 512E 4EDF 8A9D 9C4C 4414 BAAD 7DF4 BF7C 1A09
B867
```

bob esponja (minha chave aleatoria é esta, isso é confidencial) <bob@esponja.com.br>

Você tem certeza de que quer assinar esta  
chave com sua chave "bob esponja (minha chave primaria é essa) <bob@esponja.com>"  
(865777DD)

Realmente assinar? (s/N) s

Você precisa de uma frase secreta para destravar a chave secreta do usuário: "bob esponja (minha  
chave primaria é essa) <bob@esponja.com>"  
1024-bit DSA chave, ID 865777DD, criada 2009-10-18

Comando> sign

"bob esponja (minha chave aleatoria é esta, isso é confidencial) <bob@esponja.com.br>" já  
foi assinado pela chave 865777DD  
Nada para assinar com a chave 865777DD

## **Passo 4 Cifrando e decifrando com sua chaves.**

Para cifrar qualquer arquivo, basta fazer :

```
gpg -e -u "nome do usuario de origem" -r "nome do usuario de destino" arquivo
```

**ex:**

```
gpg -e -u "bob" -r "alice" bob.txt
```

**E para decifrar qualquer arquivo, basta fazer :**

```
gpg -d nomedoarquivo.gpg
```

*é necessário a frase secreta.*

```
gpg -d bob.txt.gpg
```

```
root@credson-laptop:/home/credson# nano bob.txt
root@credson-laptop:/home/credson# gpg -e -u "bob" -r "alice" bob.txt
root@credson-laptop:/home/credson# gpg -d bob.txt.gpg
```

Você precisa de uma frase secreta para destravar a chave secreta do usuário: "maria alice (esse comentario serve pera gerar minha chave aleatoria) <mariaalice@hotmail.com>" 2016-bit ELG-E chave, ID 074910F0, criada 2009-10-18 (ID principal da chave CAAB171B)

```
gpg: gpg-agent não está disponível nesta sessão
gpg: criptografado com 2016-bit ELG-E chave, ID 074910F0, criado 2009-10-18
"maria alice (esse comentario serve pera gerar minha chave aleatoria)
<mariaalice@hotmail.com>"
esse é meu testo legivel
```

### ***Comandos extras:***

***fonte: <http://www.kernelhacking.com/rodrigo/docs/gnupg.txt>***

Dicas GnuPG

Para criar uma chave:

```
gpg --gen-key
```

Para listar chaves publicas:

```
gpg --list-keys
```

Para listar chaves particulares:

```
gpg --list-secret-keys
```

Para carregar uma chave particular:

```
gpg --import --allow-secret-key-import
```

Para carregar uma chave publica:

```
gpg --import
```

Para imprimir uma chave particular em um arquivo como ASCII (usado para carregar depois, etapa anterior):

```
gpg --export-secret-key -a -o secret.key
```

Para imprimir uma chave publica em um arquivo como ASCII (usado para carregar depois, etapa anterior):

```
gpg --export -a -o public.key
```

Para apagar uma chave publica:

```
gpg --delete-key
```

Para apagar uma chave particular:

```
gpg --delete-secret-key
```

Para criptografar um arquivo:

```
gpg -e -r
```

Para assinar um arquivo:

```
gpg -s -r
```

Para criptografar e assinar um arquivo:

```
gpg -es -r
```

Para descriptografar um arquivo:

```
gpg -o -d -r
```

Para criptografar e mandar por email um arquivo:

```
gpg -e -a -r
```

Para receber uma chave publica do servidor GnuPG:

```
gpg -recv-keys
```

Para enviar uma chave publica para o servidor GnuPG:

```
gpg -send-keys
```

Referencias:

**<http://www.gnupg.org>**

**<http://irtfweb.ifa.hawaii.edu/~lockhart/gpg/gpg-cs.html>**

**<http://www.eupodiatamatando.com>**