

David Grunheidt Vilela Ordine - 16202253

## **INE5429 - 2021.2 - Trabalho PGP**

Universidade Federal de Santa Catarina  
Departamento de Informática e Estatística  
Ciência da Computação

Florianópolis  
2021

# Sumário

<b>Sumário</b>	<b>2</b>
<b>1 CRIAR CERTIFICADO</b>	<b>3</b>
<b>2 REVOGAR UM CERTIFICADO</b>	<b>4</b>
<b>3 ASSINAR CERTIFICADO DE TERCEIRO</b>	<b>5</b>
<b>4 DEMAIS PERGUNTAS</b>	<b>6</b>
4.1 O que é o anel de chaves privada?	6
4.2 Qual a diferença entre assinatura local e no servidor?	6
4.3 O que é e como é organizado o banco de dados de confiabilidade?	6
4.4 O que são e para que servem as sub-chaves?	7
4.5 Foto em certificado GPG	7
4.6 O que é preciso para criar e manter um servidor de chaves GPG, sincronizado com os demais servidores existentes?	8
4.7 como tornar sigiloso um arquivo usando o GPG?	8

# 1 Criar certificado

Inicialmente, é necessário instalar o GPG. Como o autor deste trabalho usa um notebook com sistema operacional *macOS*, foi utilizado um instalador para fins de simplificação, o qual está disponível em <https://gpgtools.org/>. Após a instalação, para gerar os certificados basta executar o comando abaixo, o qual irá pedir algumas informações como nome e email. Quando finalizado, as chaves estarão disponíveis nos diretórios informados no terminal.

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

Pra fazer o *backup* da chave privada, é necessário executar o comando abaixo, onde `<key_id>` é o identificador da chave (**CAE9CEA9**). A chave será salva no arquivo *private-key* no diretório em que o terminal estava quando o comando foi executado.

```
$ gpg --export-secret-keys --armor <key_id> > private-key
```

Já para publicar a chave publica em um repositório PGP, é necessário rodar o comando abaixo, onde `<server>` é o servidor que será enviado, no caso deste trabalho, o da rnp ([keyserver.cais.rnp.br](http://keyserver.cais.rnp.br)).

```
$ gpg --keyserver <server> --send-keys <key_id>
```

A chave foi enviada com sucesso e pode ser encontrada em <http://keyserver.cais.rnp.br> pesquisando por 0xCAE9CEA9.



Figura 1 – Chave já no repositório pgp da RNP

## 2 Revogar um certificado

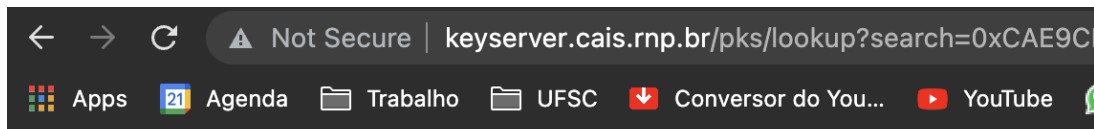
Para criar um certificado de revogação, basta executar o comando abaixo, o qual irá salvar o certificado no arquivo *revoke-key* no diretório onde o terminal se encontra. Serão feitas algumas perguntas e a senha da chave será pedida também.

```
$ gpg --output revoke-key --gen-revoke <key_id>
```

Para mover este arquivo para o repositório PGP, basta executar:

```
$ gpg --import revoke-key
```

```
$ gpg --keyserver <server> --send-keys <key_id>
```



### Search results for '0xcae9cea9'

Type	bits/keyID	Date	User ID
pub	3072R/ <a href="#">CAE9CEA9</a>	2022-03-21	*** KEY REVOKED *** [not verified] <a href="#">David &lt;davidordine98@gmail.com&gt;</a> Fingerprint=376B 54D6 AEF8 AF81 2906 C283 5D54 F21F CAE9 CEA9

Figura 2 – Chave revogada

### 3 Assinar certificado de terceiro

Para assinar o certificado de uma pessoa, é necessário adicionar sua chave no anel de chaves. Após isso, basta assinar a chave e enviá-la para o repositório PGP remoto. Para o exemplo foi usado uma outra chave criada em um computador diferente. Os comandos abaixo realizam o que foi descrito acima, onde *<key\_id>* é o id da outra chave (**B671D826**).

```
$ gpg --keyserver <server> --recv-keys <key_id>
$ gpg --sign-key <key_id>
$ gpg --keyserver <server> --send-keys <key_id>
```

Search results for '0x12c8353cb671d826'

	Type	bits/keyID	cr. time	exp time	key expir
pub	2048R/	B671D826	2022-03-22		uid <a href="mailto:alunox@ufsc.com.br">alunox@ufsc.com.br</a>
sig	sig	B671D826	2022-03-22		[selfsig]
sig	sig	6649DD12	2022-03-22		David <daviddavid@gmail.com>

Figura 3 – Nova chave assinada no repositório rnp

Para revogar a assinatura basta executar:

```
$ gpg --edit-key <key_id>
```

Será aberto um terminal, onde é necessário digitar o comando da primeira linha abaixo para que sejam informados alguns dados sobre a revogação. Ao final do processo, é necessário executar o comando da segunda linha abaixo para salvar a ação. O ultimo comando é responsável por enviar a chave para o servidor.

```
$ revsig
$ save
$ gpg --keyserver <server> --send-keys <key_id>
```

Search results for '0x12c8353cb671d826'

	Type	bits/keyID	cr. time	exp time	key expir
pub	2048R/	B671D826	2022-03-22		uid <a href="mailto:alunox@ufsc.com.br">alunox@ufsc.com.br</a>
sig	sig	B671D826	2022-03-22		[selfsig]
sig	sig	6649DD12	2022-03-22		David <daviddavid@gmail.com>
sig	revok	6649DD12	2022-03-22		David <daviddavid@gmail.com>

Figura 4 – Chave de outro usuário revogada

## 4 Demais perguntas

### 4.1 O que é o anel de chaves privada?

O anel de chaves privadas contem as chaves do usuário que são utilizadas para aplicações GPG. Encriptar um arquivo ou assinar um documento são ações que usarão **subkeys** específicas, as quais ficam armazenadas no anel de chaves privadas. Abaixo mostra-se o comando para listar as chaves privadas e o resultado do comando executado. Nota-se que elas ficam no diretório `/Users/davidordine/.gnupg/pubring.kbx` e o arquivo que as armazena só permite leitura.

```
[davidordine@PK60-DORDINE Trab III - PGP % gpg --list-secret-keys
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid: 1  signed: 0  trust: 0-, 0q, 0n, 0m, 0f, 1u
/Users/davidordine/.gnupg/pubring.kbx
-----
sec   rsa3072 2022-03-21 [SC] [revoked: 2022-03-21]
      376B54D6AEFEAF812906C2835D54F21FCAE9CEA9
uid   [ revoked] David <davidordine98@gmail.com>

sec   rsa3072 2022-03-22 [SCEA]
      A99215AEBCE96199926CE7E6416C76606649DD12
ssb   rsa3072 2022-03-22 [E] [expires: 2024-03-21]
```

Figura 5 – Chave de outro usuário revogada

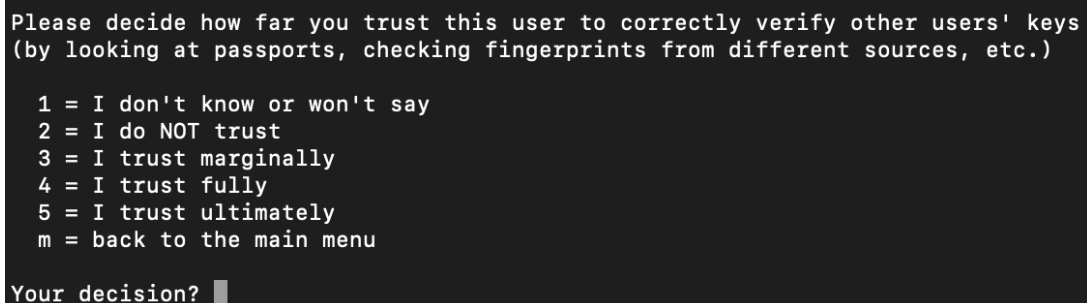
### 4.2 Qual a diferença entre assinatura local e no servidor?

Sem o uso de um servidor de certificados, se certo usuário X assina o certificado de outro usuário Y, o usuário X tem que mandar o certificado de Y assinado para Y. Quando Y recebe este novo certificado assinado é necessário o enviar para todos os outros usuários que precisarem. Através dos servidores, o usuário X só precisa assinar o certificado de Y e mandar para o servidor. Outros usuários conseguem atualizar o certificado de Y, que agora foi assinado por X, através de uma simples operação de *fetch* no servidor.

### 4.3 O que é e como é organizado o banco de dados de confiabilidade?

O banco de dados de confiabilidade tem como responsabilidade guardar as informações de confiança que certo usuário possui sobre outras chaves. Para alterar qual o nível de confiança associado a uma certa chave, basta executar os comandos abaixo.

```
$ gpg --edit-key <key_id>
$ trust
```

A screenshot of a terminal window showing the GPG trust level selection prompt. The text is as follows:

```
Please decide how far you trust this user to correctly verify other users' keys
(by looking at passports, checking fingerprints from different sources, etc.)

 1 = I don't know or won't say
 2 = I do NOT trust
 3 = I trust marginally
 4 = I trust fully
 5 = I trust ultimately
 m = back to the main menu

Your decision? █
```

Figura 6 – Escolha do nível de confiança

## 4.4 O que são e para que servem as sub-chaves?

No anel existem várias chaves privadas. Dentre elas:

1. *Master key*: Chave que tem como função identificar o usuário. Usada para assinar email e nome do usuário nos certificados.
2. *Subkeys*: Outras chaves no anel. São utilizadas para assinar e encriptar dados reais. A *master key* assina as *subkeys* para informar que estas pertencem ao usuário.

Com este esquema, a ideia é tornar o gerenciamento de chaves mais fácil. É possível substituir as *subkeys* e a *Master key* fica bem menos exposta.

## 4.5 Foto em certificado GPG

Para enviar uma imagem para um certificado gpg é necessário executar os comandos abaixo.

```
$ gpg --edit-key <key_id>
$ addphoto
$ /Users/davidordine/Desktop/meme.png
$ save
$ gpg --keyserver <server> --send-keys <key_id>
```



Figura 7 – Imagem enviada ao certificado

Para verificar se a foto foi associada ao certificado, basta executar:

```
$ gpg --list-options show-photos --list-keys
```

```
pub  rsa3072 2022-03-22 [SCEA]  
      A99215AEBCE96199926CE7E6416C76606649DD12  
uid      [ultimate] [jpeg image of size 7183]  
sub  rsa3072 2022-03-22 [E] [expires: 2024-03-21]
```

Figura 8 – Imagem associada ao certificado

## 4.6 O que é preciso para criar e manter um servidor de chaves GPG, sincronizado com os demais servidores existentes?

Primeiramente, é preciso ter acesso aos *dumps* das chaves dos outros servidores. Assim, é possível adicionar esta base de chaves no servidor que está sendo desenvolvido. Na realidade, estes *dumps* citados acima são de difícil acesso, visto que o número de servidores que liberam seus *dumps* de forma gratuita é baixo. Além disso, quando são disponibilizados, há toda uma burocracia por trás.

## 4.7 como tornar sigiloso um arquivo usando o GPG?

Primeiro é preciso importar a chave pública do outro usuário para o computador. Após isso, é possível encriptar um arquivo com a chave pública importada. Os dois comandos abaixo exemplificam essas ações. Foi importado a chave pública com *key id* **B671D826**. O arquivo codificado será salvo como *file.gpg*.



```
$ gpg --keyserver keyserver.cais.rnp.br --recv 6649DD12  
$ gpg --output file.gpg --encrypt --recipient 6649DD12 file
```

Após codificado, basta enviar o arquivo para a pessoa que possui a chave privada da chave publica usada para codificar. Para decodificar a mensagem, basta executar o comando abaixo, o qual irá pedir a senha da sua chave privada.

```
$ gpg --output file_decoded --decrypt file.gpg
```

```
[davidordine@PK60-DORDINE Trab III - PGP % gpg --output file_decoded --decrypt file.gpg  
gpg: encrypted with 3072-bit RSA key, ID 81D0C6C83E16C8B3, created 2022-03-22  
"David <daviddavid@gmail.com>"  
[File 'file_decoded' exists. Overwrite? (y/N) Y  
[davidordine@PK60-DORDINE Trab III - PGP % cat file_decoded  
arquivo malucão  
davidordine@PK60-DORDINE Trab III - PGP % █
```

Figura 9 – Arquivo decodificado