

88% (B+) PGP (85%): algumas imprecisões; mostram as sub chaves, e comentam-nas de forma adequada; configuração do keyserver e comuni

Catarina da Cunha Malheiro da Silva Pereira
(PG53733)
Inês Cabral Neves (PG53864)
Leonardo Dias Martins (PG53996)
Miguel José Mendes Gonçalves (PG54101)
Rui Fernando dos Santos Barbosa (A89370)

TP3 - Chaves de Cifra, Certificados e o PGP

Relatório Prático de Cibersegurança
Mestrado em Engenharia
Telecomunicações e Informática

Trabalho efetuado sob a orientação de:
Professor Doutor Henrique Manuel Dinis Santos

Índice

Índice de Figuras	iii
Lista de Acrónimos	v
1 Introdução	1
2 Revisão da Literatura	2
3 Opção PGP	3
3.1 Gestão de chaves	3
3.2 Enviar e receber mensagens seguras	10
3.3 Proteger documentos locais	17
4 Opção X509	19
4.1 Gestão de chaves	19
4.2 Enviar e receber mensagens seguras	25
5 Conclusão	28
Referências Bibliográficas	29

Índice de Figuras

1	Ambiente Kleopatra ao iniciar.	3
2	Criação de Chave no Kleopatra.	3
3	Configuração avançada.	4
4	Confirmação da criação do certificado.	4
5	Assistente de Criação de pares de chaves.	5
6	Certificações disponíveis no Kleopatra.	5
7	Detalhes das Subchaves.	5
8	Criação de par de chaves.	6
9	Criação de par de chaves (continuação).	6
10	Pedido de uma Certification Authority.	6
11	Configurar servidor de chaves OpenPGP.	7
12	Estruturação do servidor.	7
13	Seleção do servidor e da porta a utilizar.	7
14	Exportação no servidor.	7
15	Confirmação da exportação no Kleopatra.	7
16	Resultados da pesquisa por endereço e-mail hsantos@dsi.uminho.pt.	8
17	Resultados da pesquisa pelo nome “Henrique Santos”.	8
18	Chave publica enviada por e-mail.	9
19	Certificação da chave pública de um membro do grupo.	9
20	Certificação da chave pública de um membro do grupo (continuação).	10
21	Certificação terminada com sucesso.	10
22	Adicionar chave OpenPGP ao Thunderbird (1).	10
23	Adicionar chave OpenPGP ao Thunderbird (2).	11
24	Chave importada com sucesso (1).	11
25	Chave importada com sucesso (2).	11
26	Chave adicionada.	11
27	Envio de chave pública PGP.	12
28	Importar chave pública PGP de outros (1).	12
29	Importar chave pública PGP de outros (2).	12
30	Importar chave pública importada com sucesso.	13
31	Gestor de chaves OpenPGP.	13
32	Envio de email seguro da Inês para o Leonardo.	13
33	Receção do email seguro da Leonardo para a Inês.	14
34	Extração do Certificado de Revogação.	14
35	Certificado de Revogação.	14
36	Chave Revogadas.	15
37	Importação do Certificado de Revogação.	15
38	Chave revogada no Thunderbird.	15
39	Receção do email encriptado do Leonardo para a Inês.	16
40	Email com certificado de revogação em anexo.	16
41	Chave revogada no gestor de chaves do Leonardo.	17
42	Tentativa de enviar email para a Inês.	17
43	Opção para assinar/cifrar a pasta.	17
44	Escolha do certificado e pasta.	18
45	Confirmação da cifra.	18
46	Verificação da Versão do OpenSSL.	19
47	Criação de um novo par de chaves.	19
48	Verificação das chaves criadas.	19
49	Verificação do estado da chave privada.	19

Índice de Figuras

50	Criação do certificado.	20
51	Verificação do estado de pedido de certificado.	20
52	Criação do certificado auto-assinado.	21
53	Criação do certificado auto-assinado.	21
54	Criação de um repositório remoto.	22
55	Criação da diretorias.	22
56	Criação da base de dados.	22
57	Pedido de autenticação de certificado.	22
58	Criação de um certificado CA auto assinado.	22
59	Criação do certificado CA certificado pela Root CA	23
60	Criação de um pedido de certificado de email.	23
61	Aceitação do pedido por parte da Signing CA e criação do certificado.	23
62	Criação do ficheiro PKCS12	24
63	Verificação do ficheiro PKCS12 (1).	24
64	Verificação do ficheiro PKCS12 (2).	24
65	Import do certificado.	25
66	Import do certificado da autoridade certificadora criada.	25
67	Seleção dos certificados para assinar e decifrar.	25
68	Email a confirmar a encriptação.	26
69	Seleção dos certificados para assinar e decifrar.	26
70	Revogação do Certificado.	26
71	Obtenção da CRL.	27
72	Lista de CRL.	27

Acrónimos

AES Advanced Encryption Standard.

CAs Autoridades de Certificação.

CN Common Name.

CRL Certificate Revocation List.

DES Data Encryption Standart.

OCSP Online Certificate Status Protocol.

PEM Privacy Enhanced Mail.

PKI Public Key Infrastructure.

UC Unidade Curricular.

1 Introdução

Este relatório é parte integrante da Unidade Curricular (UC) de Cibersegurança do 2º semestre do 1º ano do Mestrado Integrado em Engenharia de Telecomunicações e Informática. Foi elaborado em resposta a um problema apresentado pelo docente.

O relatório **aborda duas abordagens de** comunicação segura para a troca de emails: PGP e x509, divididas em duas partes distintas. A primeira parte analisa a utilização do PGP, incluindo a gestão de chaves, o envio e receção de mensagens seguras, e a proteção de documentos locais. Na segunda parte, é explorada a utilização do x509, com foco na gestão de chaves e no envio e receção de mensagens seguras.

2 Revisão da Literatura

a criptografia não se esgota nos algoritmos de cifra. E a cifra simétrica/assim

A criptografia pode ser dividida em **dois tipos principais**: criptografia de chave simétrica e criptografia de chave pública ou assimétrica [1].

- **Criptografia de Chave Simétrica:**

A criptografia de chave simétrica utiliza a mesma chave para cifrar e decifrar informações. Algoritmos de chave simétrica, como o Data Encryption Standard (DES) e Advanced Encryption Standard (AES), empregam essa abordagem, onde a mesma chave é compartilhada entre as partes envolvidas na comunicação.

- **Criptografia de Chave Pública ou Assimétrica:**

Por outro lado, a criptografia de chave pública, também conhecida como criptografia assimétrica, utiliza um par de chaves matematicamente relacionadas: uma chave pública e uma chave privada. A mensagem é cifrada com a chave pública e decifrada com a chave privada. Isso permite a distribuição livre da chave pública, enquanto a chave privada permanece em posse exclusiva do proprietário [2].

ou vice versa

A criptografia de chave pública possibilita a comunicação segura entre partes sem a necessidade de um canal seguro para transmitir chaves secretas, tornando-a uma abordagem mais flexível e segura em comparação com a criptografia simétrica tradicional.

Os certificados digitais e a Public Key Infrastructure (PKI) são essenciais para **garantir a segurança e autenticidade** das comunicações online [3, 4].

mais para dar co

quando muito, chaves privadas em assinaturas!

- Os certificados digitais **utilizam chaves públicas para validar a identidade dos detentores**, permitindo a comunicação segura, integridade de dados e autenticação. As chaves privadas são cruciais para a criptografia assimétrica, garantindo a segurança das transmissões de dados confidenciais em conexões seguras SSL/TLS [3].
- A PKI é o sistema responsável por criar, gerir, armazenar, distribuir e revogar os Certificados de Chave Pública, sendo fundamental para estabelecer a confiança nas interações online. As Autoridades de Certificação (CAs) desempenham um papel crucial na emissão e gestão dos certificados digitais, atuando como agentes de confiança na PKI. A confiança na PKI é baseada na validação das identidades dos usuários e na integridade dos certificados emitidos pelas CAs.

Os certificados digitais e a PKI são **elementos fundamentais para garantir a autenticidade, integridade e confidencialidade** das comunicações online, protegendo contra atividades fraudulentas e assegurando a confiança nas transações digitais.

Escrito sem grande rigor

3 Opção PGP

Para gerar um certificado PGP, foi utilizado o gestor de certificados Kleopatra num ambiente Windows, Figura 1. Kleopatra é um software gerador de certificados para GnuPG que armazena todas as chaves e certificados OpenPGP no dispositivo do utilizador, facilitando a resolução dos exercícios propostos.



Figura 1: Ambiente Kleopatra ao iniciar.

3.1 Gestão de chaves

Na Figura 2 é demonstrado o processo de criação uma chave PGP, onde se atribuiu o nome e e-mail à mesma.

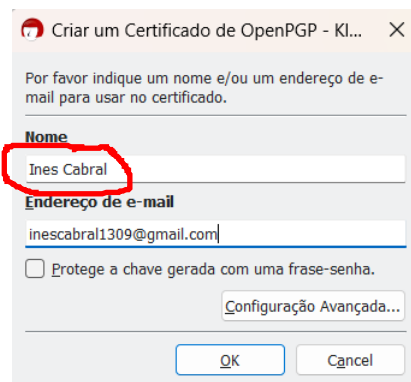


Figura 2: Criação de Chave no Kleopatra.

Na configuração avançada, Figura 3, do par de chaves, foi escolhido o algoritmo RSA com 1024/2048 bits, sem data-limite de validade e mantendo a lista de algoritmos para a cifra. Este tipo de chave RSA é assimétrico, com uma chave pública e uma chave privada utilizadas pelo destinatário e remetente no processo de criptografia.

3.1. GESTÃO DE CHAVES

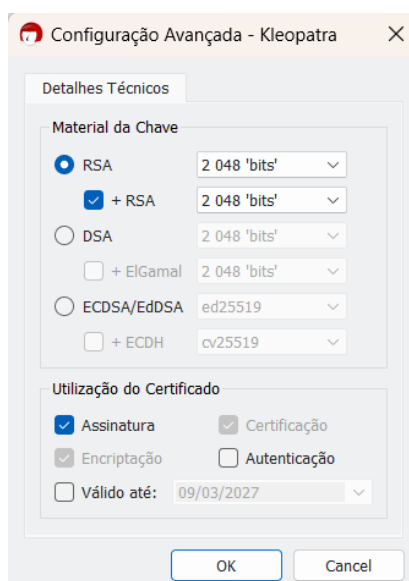


Figura 3: Configuração avançada.

A *passphrase* escolhida pelo grupo foi “cibergrupo5”, selecionada por ser simples e fácil de memorizar. Esta *passphrase* deve ser inserida sempre que a chave privada for utilizada. A confirmação da criação do certificado está representada na Figura 4.

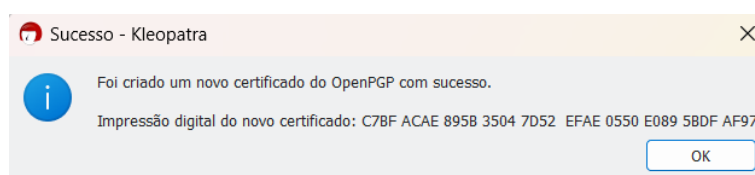


Figura 4: Confirmação da criação do certificado.

Como se pode observar pela Figura 5 e a Figura 6, os atributos mais significativos da assinatura e da chave são:

- ID da chave.
- Email: inescabral1309@gmail.com.
- Data de criação: 09 de março de 2024.
- Tipo de chave escolhido: OpenPGP.
- Tempo: Ilimitado
- Impressão digital: C7BF ACAE 895B 3504 7D52 EFAE 0550 E089 5BDF AF97



Esta impressão digital é um *hash* da chave pública e é gerada através de uma relação estabelecida entre a assinatura e uma chave privada. De forma a obter uma verificação, é aplicada a chave pública que deverá corresponder à chave privada que gerou a assinatura.

3.1. GESTÃO DE CHAVES

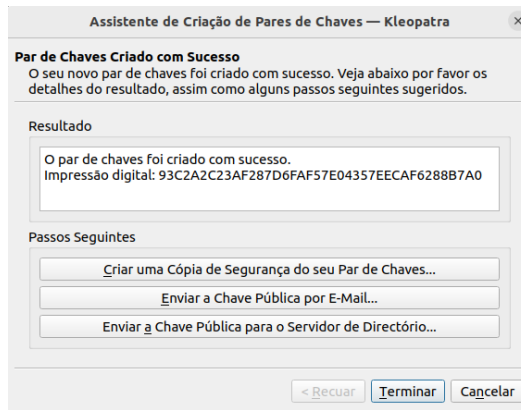


Figura 5: Assistente de Criação de pares de chaves.

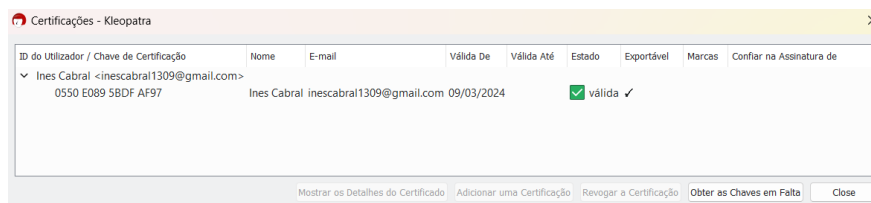
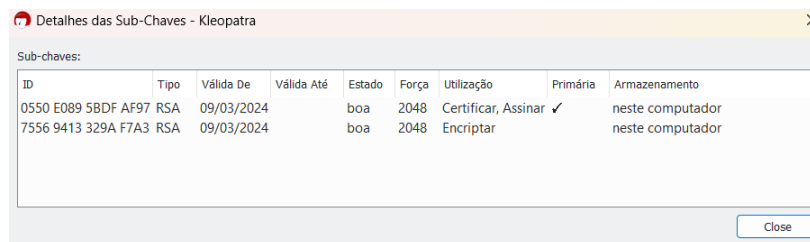


Figura 6: Certificações disponíveis no Kleopatra.

Na Figura 7, são apresentadas duas sub chaves: uma destinada à certificação e assinatura, e outra exclusivamente para encriptação [5]. Ambas as sub-chaves possuem componentes de chave pública e privada. A primeira sub chave utiliza a chave privada para assinar digitalmente e a chave pública para verificação da autenticidade das mensagens assinadas. Por outro lado, a segunda sub chave utiliza a chave privada para desencriptar dados cifrados, os quais foram encriptados com a chave pública previamente compartilhada pela entidade remetente. **A gestão da segunda sub-chave é realizada pela chave mestra, responsável pelo backup e revogação das sub-chaves. É importante destacar que a segunda sub-chave é derivada da chave primária.**



Derivada, como?E o que

Figura 7: Detalhes das Subchaves.

Na Figura 8 e na Figura 9 são apresentados os comandos executados na linha de comandos de um ambiente Windows para a criação de novas sub-chaves. O status da chave é exibido como revogado, no entanto, foi possível adicionar a nova sub-chave.

3.1. GESTÃO DE CHAVES

```
C:\Users\Ines Cabral> gpg --full-generate-key
gpg (GnuPG) 2.4.4; Copyright (C) 2024 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Selecione o tipo de chave desejado:
  (1) RSA e RSA
  (2) DSA e Elgamal
  (3) DSA (apenas de assinar)
  (4) RSA (apenas de assinar)
  (9) ECC (de assinar e cifrar) *pré-definição*
  (10) ECC (apenas de assinar)
  (14) Chave do cartão existente
Sua opção? 4
As chaves RSA podem estar entre 1024 e 4096 bits de comprimento.
Qual tamanho de chave você quer? (3072) 2048
O tamanho de chave pedido é 2048 bits
Especifique quando a chave expira.
  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
Quando a chave expira? (0) 0
A chave não expira de forma alguma
Isto está correto? (s/N) s

O GnuPG precisa construir uma ID de utilizador para identificar sua chave.

Nome verdadeiro: Ines Cabral
Endereço de email: inescabral1309@gmail.com
Comentário:
Você selecionou este USER-ID:
  "Ines Cabral <inescabral1309@gmail.com>"
```

Figura 8: Criação de par de chaves.

```
Alterar (N)ome, (C)omentário, (E)ndereço, ou (O)k/(S)air? 0
Precisamos gerar muitos bytes aleatórios. É uma boa ideia realizar outra
atividade (escrever no teclado, mover o rato, usar os discos) durante a
geração dos números primos; isto dá ao gerador de números aleatórios
uma hipótese maior de ganhar entropia suficiente.
gpg: certificado de revogação armazenado como 'C:\Users\Ines Cabral\AppData\Roaming\gnupg\openpgp-revocs.d\B6CFE5
8CAFDA451611695FF0CB88E77386FAA257.rev'
chaves pública e privada criadas e assinadas.

Repare que esta chave não pode ser usada para cifração. Poderá querer usar
o comando "--edit-key" para gerar uma subchave com esta finalidade.
pub  rsa2048 2024-03-10 [SC]
     B6CFE58CAFDA451611695FF0CB88E77386FAA257
uid                               Ines Cabral <inescabral1309@gmail.com>
```

Figura 9: Criação de par de chaves (continuação).

Para obter um certificado do tipo X509 com o par de chaves previamente criado, é essencial solicitar à CA, conforme ilustrado na Figura 10. Apesar da criação do par de chaves, o pedido à CA para a emissão do certificado não foi finalizado.

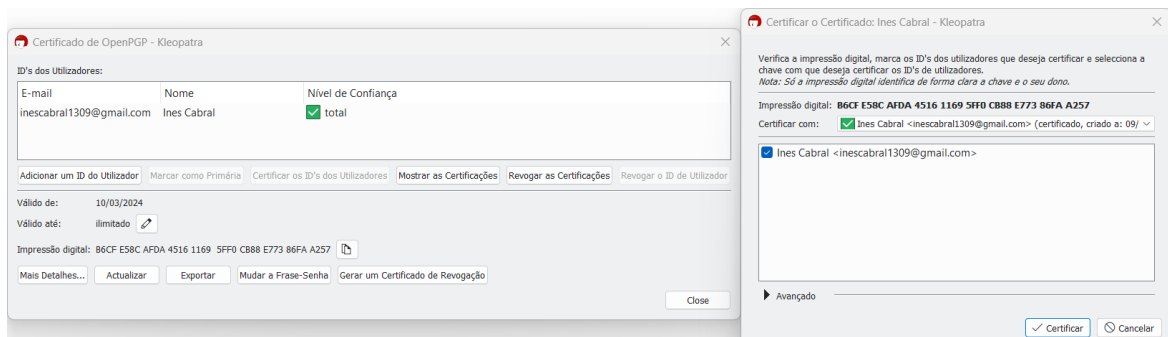


Figura 10: Pedido de uma Certification Authority.

Para este procedimento, foi essencial configurar a aplicação para o servidor PGP, utilizando o "hkps://pgpkeys.mit.edu", onde a chave pública será armazenada e permitirá a sua pesquisa no navegador, Figura 11.

3.1. GESTÃO DE CHAVES

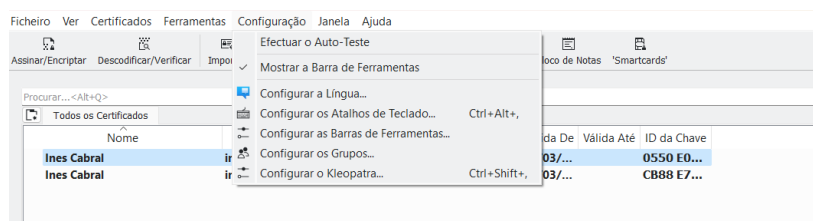


Figura 11: Configurar servidor de chaves OpenPGP.

A chave pública foi exportada para o servidor pgpkeys.mit.edu conforme demonstrado na Figura 12 e Figura 13, seguindo a configuração apresentada.

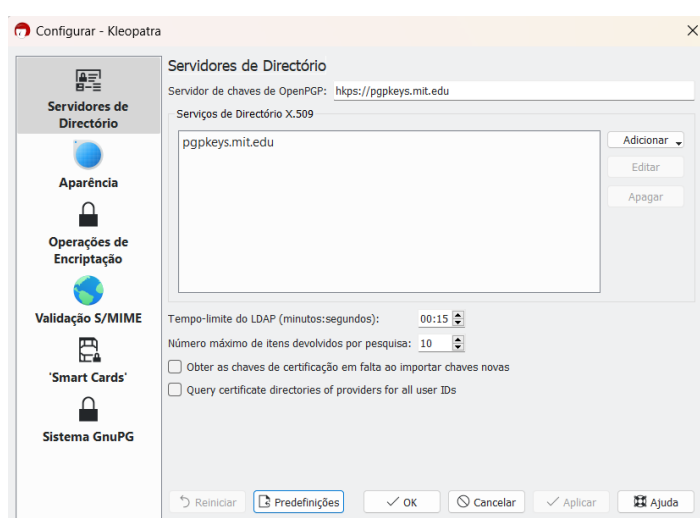


Figura 12: Estruturação do servidor.

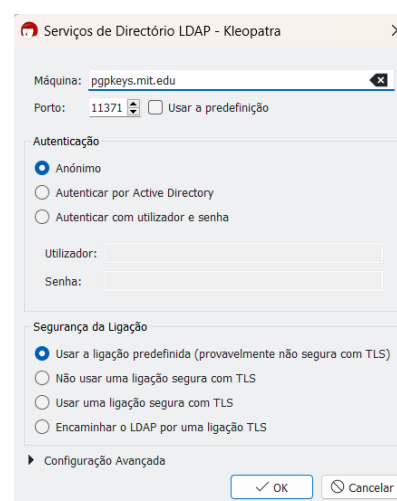


Figura 13: Seleção do servidor e da porta a utilizar.

Na Figura 14, o certificado foi exportado para o servidor hhttps://pgpkeys.mit.edu/ após configurar a aplicação para o servidor PGP. Isso permitiu a publicação da chave pública no servidor, conforme ilustrado na Figura 15.

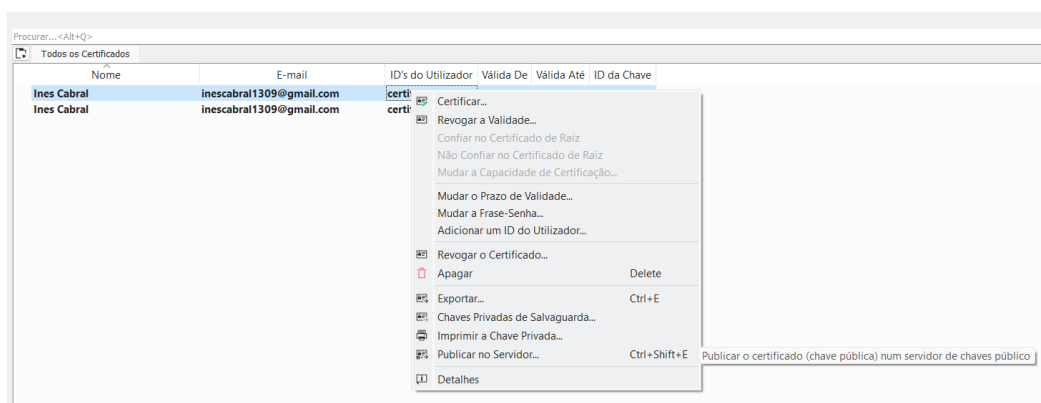


Figura 14: Exportação no servidor.

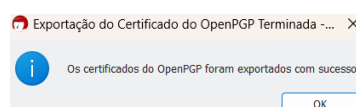


Figura 15: Confirmação da exportação no Kleopatra.

3.1. GESTÃO DE CHAVES

Após armazenar a chave pública no servidor PGP, foi realizada uma pesquisa pelas chaves públicas utilizando o endereço de e-mail (hsantos@dsi.uminho.pt) e pelo nome (Henrique Santos). Como esperado, a pesquisa pelo nome resultou em mais resultados do que a pesquisa pelo e-mail, devido à natureza única do endereço de e-mail em comparação com o nome. Essa análise foi demonstrada nas Figura 16 e Figura 17 ao consultar o servidor <http://pgpkeys.mit.edu/>.

Search results for 'uminho pt hsantos dsi'

Type	bits/keyID	Date	User ID
pub	2048R/ 18A842EA	2018-11-01	Henrique M D Santos <henrique.dinis.santos@gmail.com> HSantos <henrique.dinis.santos@dsi.uminho.pt> Henrique Santos <henrique.dinis.santos@gmail.com>
pub	2048R/ 3473AE1C	2016-09-14	Henrique Santos (Chave para uso na UM) <hsantos@dsi.uminho.pt>
pub	1024D/ 475D4617	2006-07-13	Henrique M D Santos (No) <hsantos@dsi.uminho.pt>
pub	1024D/ 3AE27210	2003-11-14	*** KEY REVOKED *** [not verified] Henrique M D Santos <hsantos@dsi.uminho.pt> Henrique M D Santos (Para uso pessoal) <henrique.dinis.santos@gmail.com> [user attribute packet]
pub	1024D/ 319D3D84	2001-06-15	Henrique Manuel Dinis dos Santos <hsantos@dsi.uminho.pt>

Figura 16: Resultados da pesquisa por endereço e-mail hsantos@dsi.uminho.pt.

Search results for 'santos henrique'

Type	bits/keyID	Date	User ID
pub	3072R/ 49EEF789	2022-02-24	Paulo Henrique dos Santos <ownerbr@gmail.com>
pub	3072R/ 26B2A788	2021-05-19	Henrique Santos <hfigueiredosantos@tecnico.ulisboa.pt>
pub	3072R/ 5DA4EEEE	2021-05-17	alexandre henrique santos grisende <alexandre.grisende@aedb.br>
pub	3072R/ DBEAD5D7	2021-05-17	alexandre henrique santos grisende <alexandre.grisende@aedb.br>
pub	2048R/ EC68DAD8	2020-04-23	joao.h.santos@layer8.pt João Henrique Santos <joao.h.santos@layer8.pt>
pub	2048R/ 5E458BDA	2020-02-18	JORGE HENRIQUE SANTOS GARCEZ <mistergarcez@hotmail.com>
pub	2048R/ 18A842EA	2018-11-01	Henrique M D Santos <henrique.dinis.santos@gmail.com> HSantos <henrique.dinis.santos@dsi.uminho.pt> Henrique Santos <henrique.dinis.santos@gmail.com>
pub	2048R/ 86F90D2B	2018-10-23	Henrique Santos <henrique.santos@inf.aedb.br>
pub	3072R/ F3C5E85D	2018-08-29	Henrique dos Santos Goulart <henrique.goulart@chaordicsystems.com>
pub	1024D/ E759B578	2018-06-12	Luiz Henrique Silva Santos <luizhenriqueeduardoss@gmail.com>
pub	2048R/ 37F1E1F6	2018-05-16	Carlos Henrique dos Santos <kc-ny@hotmail.com>

Figura 17: Resultados da pesquisa pelo nome “Henrique Santos”.

Na Figura 18, a chave pública foi exportada do Kleopatra em formato .asc, convertida para um formato .txt e enviada para o endereço de e-mail Gmail do outro membro do grupo foram passos essenciais nesse processo.

3.1. GESTÃO DE CHAVES

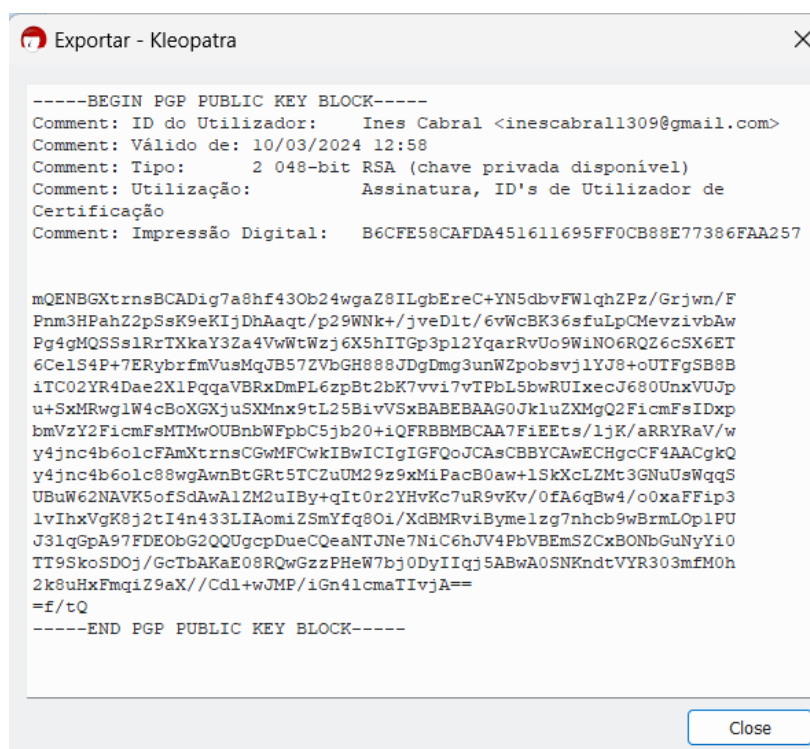


Figura 18: Chave publica enviada por e-mail.

O arquivo é **aberto automaticamente** no software e a chave é certificada, como evidenciado na Figura 19, Figura 20 e na Figura 21.



A Inês Cabral enviou a sua chave pública para a Catarina Pereira via e-mail, utilizando a técnica “drag and drop” para transferir a chave. É crucial garantir a autenticidade da chave recebida para associá-la corretamente à pessoa, evitando possíveis impostores, o que pode ser feito através de certificados digitais.

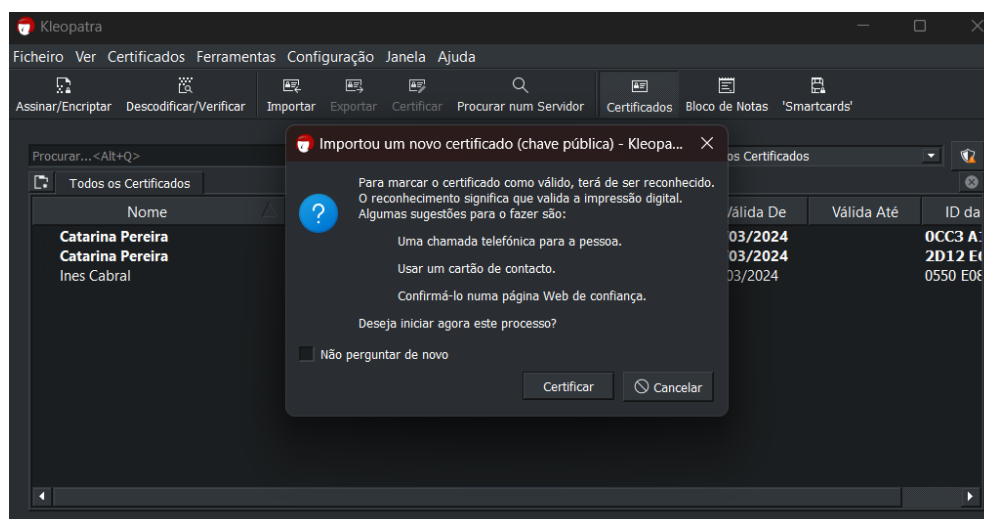


Figura 19: Certificação da chave pública de um membro do grupo.

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

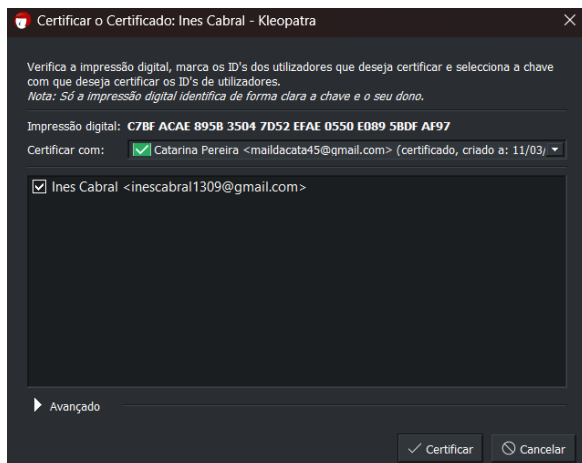


Figura 20: Certificação da chave pública de um membro do grupo (continuação).

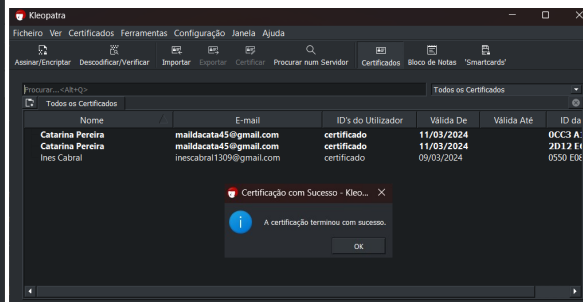


Figura 21: Certificação terminada com sucesso.

Neste ponto seria importante mostrar também o resultado obtido

3.2 Enviar e receber mensagens seguras

Pressupõem-se que os alunos do grupo importem os seus certificados para o cliente de email Thunderbird, para que através do mesmo possam trocar mensagens seguras. Começou-se por adicionar uma chave OpenPGP pessoal ao Thunderbird, conforme documentado nas figuras 22 a 22. Este procedimento foi repetido para os alunos do grupo: **Inês Cabral e Leonardo Martins**, cada um utilizado as suas próprias chaves. Ambos os alunos utilizaram este cliente em ambiente Windows.

Então não devia ser a Catarina

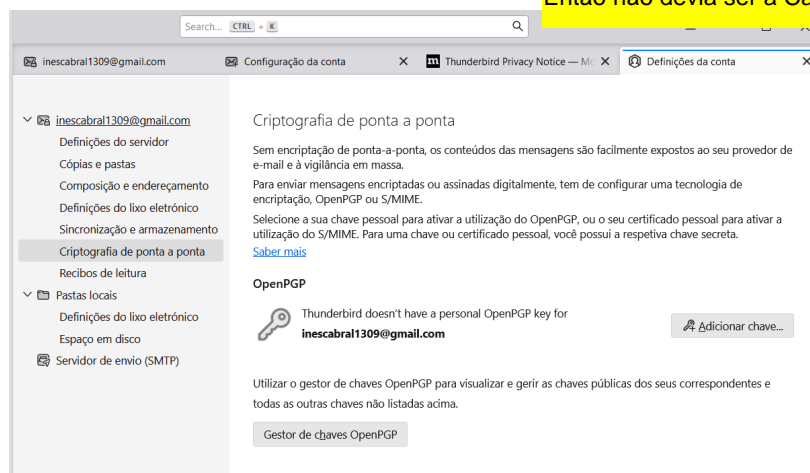


Figura 22: Adicionar chave OpenPGP ao Thunderbird (1).

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

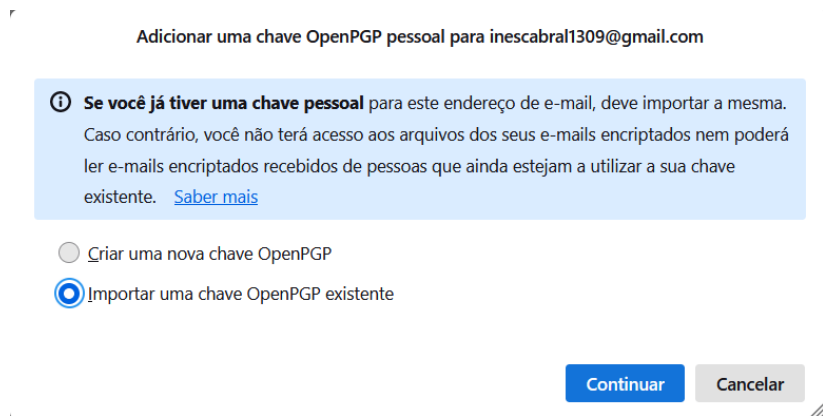


Figura 23: Adicionar chave OpenPGP ao Thunderbird (2).

As figuras 24, 25 e 26 demonstram a adição no Thunderbird de uma chave OpenPGP pessoal.

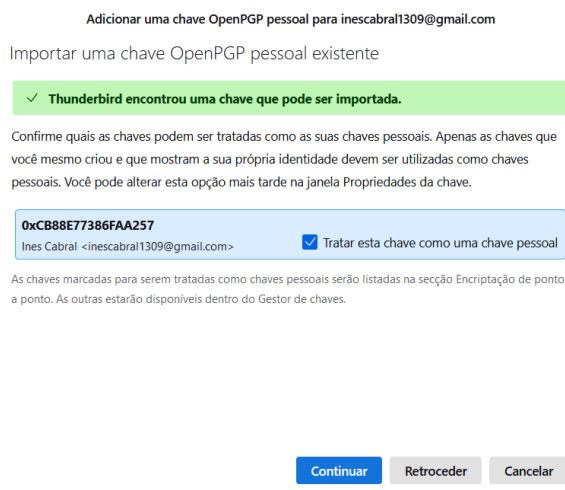


Figura 24: Chave importada com sucesso (1).

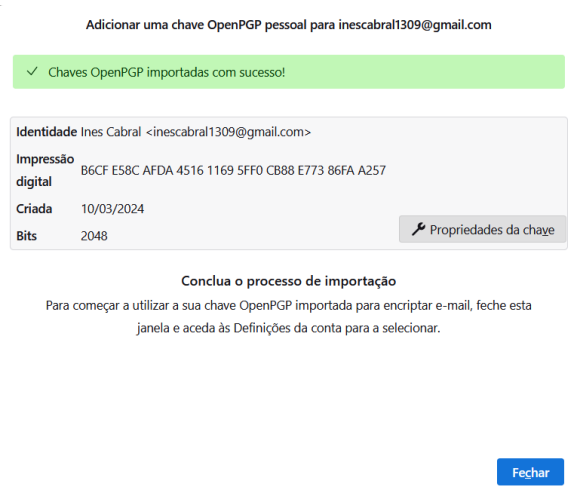


Figura 25: Chave importada com sucesso (2).

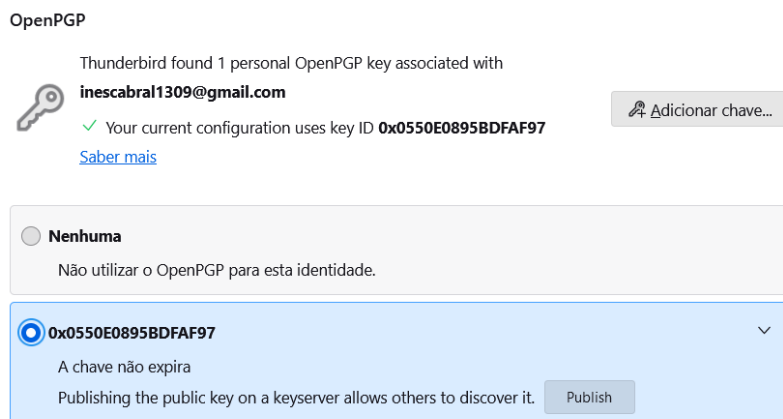


Figura 26: Chave adicionada.

Agora, é necessário inserir as chaves públicas dos destinatários para enviar emails seguros. A Inês e o Leonardo enviaram as suas chaves como anexos por email. Desta forma, ao receber a chave pública de alguém na aplicação Thunderbird, é possível adicioná-la diretamente à lista de chaves, conforme demonstrado na figura 27.

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

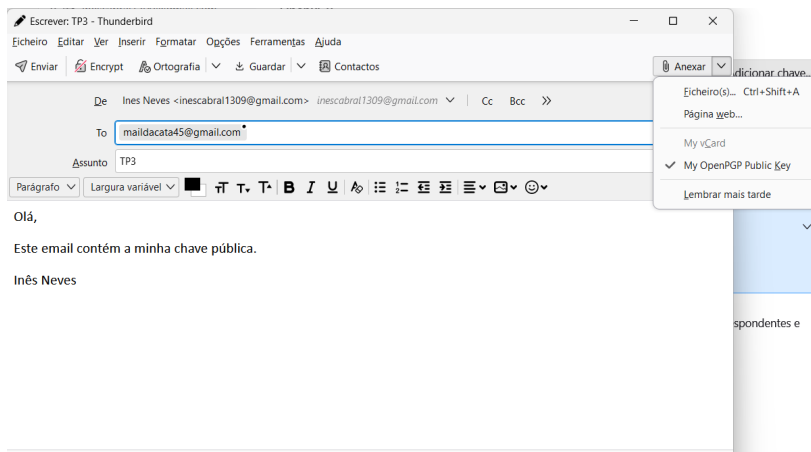


Figura 27: Envio de chave pública PGP.

Ao receber uma chave pública PGP por email, o Thunderbird permite importar a mesma, conforme os passos demonstrados nas figuras 28, 29 e 30.



Figura 28: Importar chave pública PGP de outros (1).

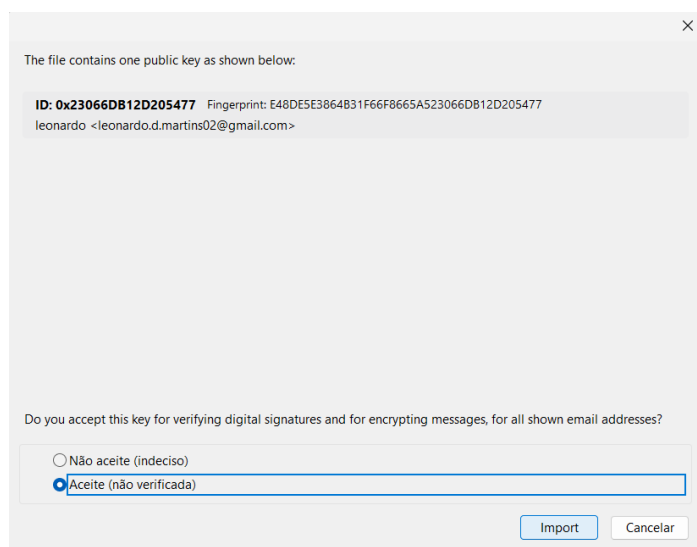


Figura 29: Importar chave pública PGP de outros (2).

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

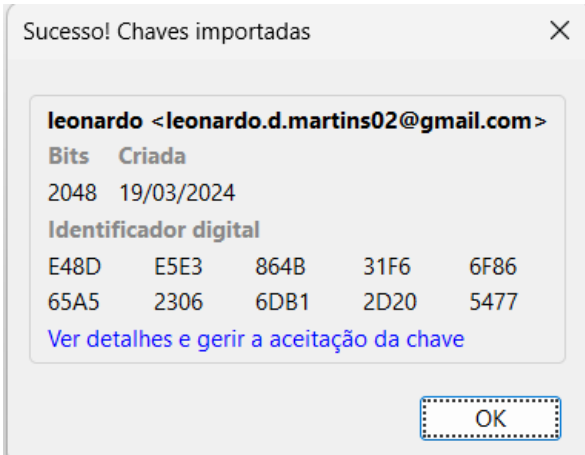


Figura 30: Importar chave pública importada com sucesso.

Como é possível observar na figura 31, a chave do Leonardo foi adicionada ao gestor de chaves da Inês.

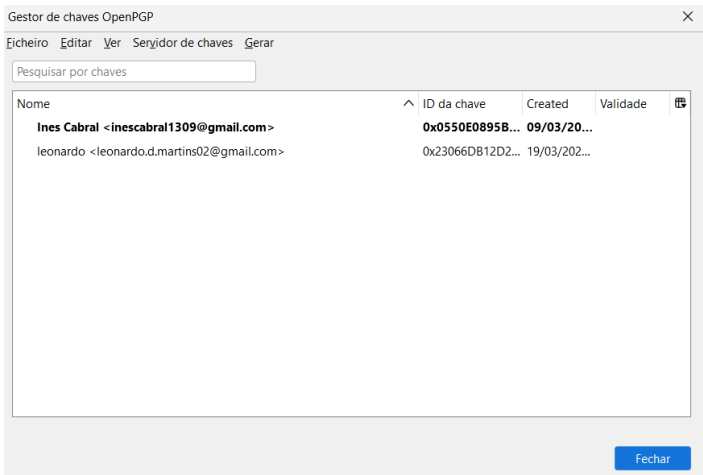
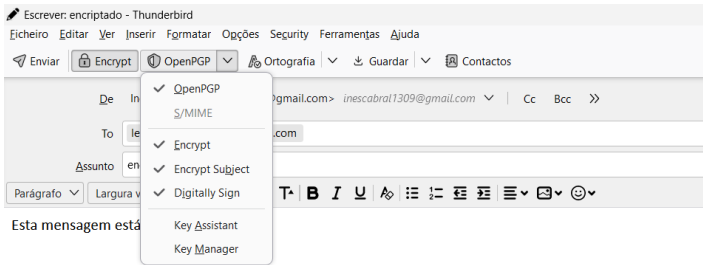


Figura 31: Gestor de chaves OpenPGP.

O Leonardo efectou os mesmos passos para a importação da chave da Inês, ou seja, estão em condições de trocar emails seguros (encriptados e assinados). Se for seleccionado o botão Encrypt, todos os parâmetros do separador Security são ativados. A chave pública também é sempre enviada em anexo. Na figura 32 é possível observar o envio de um email seguro para o Leonardo. Na figura 33 observa-se a receção do email seguro do Leonardo para a Inês, anteriormente o Leonardo também enviou para a Inês um email seguro.



evidências?!

Figura 32: Envio de email seguro da Inês para o Leonardo.

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

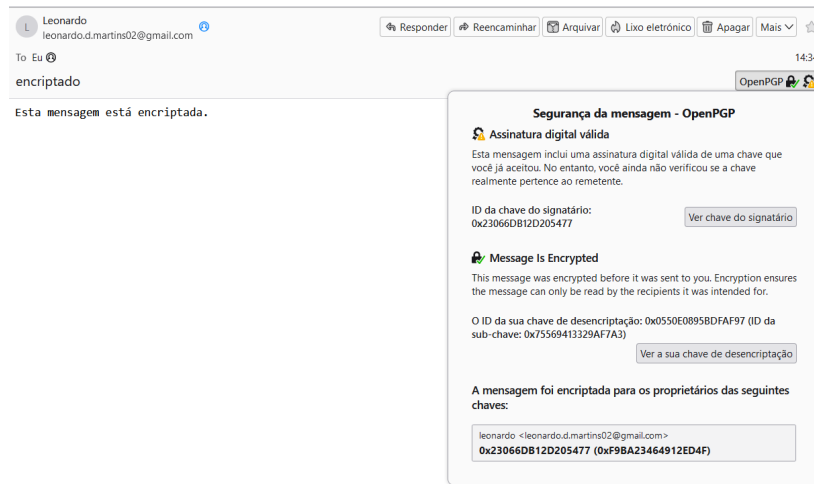


Figura 33: Receção do email seguro da Leonardo para a Inês.

É pretendido revogar os certificados criados e testar as consequências. No caso dos certificados PGP, estes foram revogados na aplicação Kleopatra e o certificado de revogação correspondente foi extraído. Esta revogação **pode ser publicada no servidor**. As figuras 34 e 36 mostram a chave revogada e a extração do certificado de revogação associado.

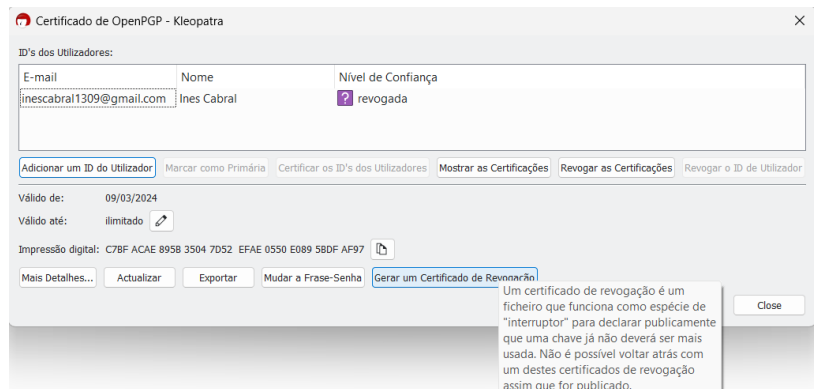


Figura 34: Extração do Certificado de Revogação.



Figura 35: Certificado de Revogação.

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

Nome	E-mail	ID's do Utilizador	Válida De	Válida Até	ID da Chave
Ines Cabral	inescabral1309@gmail.com	revogada	09/03/2024		0550 E089 5BDF AF97

Figura 36: Chave Revogadas.

Depois de obter o Certificado de Revogação, este foi adicionado ao gestor de chaves no Thunderbird e a chave que a Inês estava a utilizar deixou de ser válida para o envio de emails seguros. Este processo está demonstrado nas figuras 37 e 38.

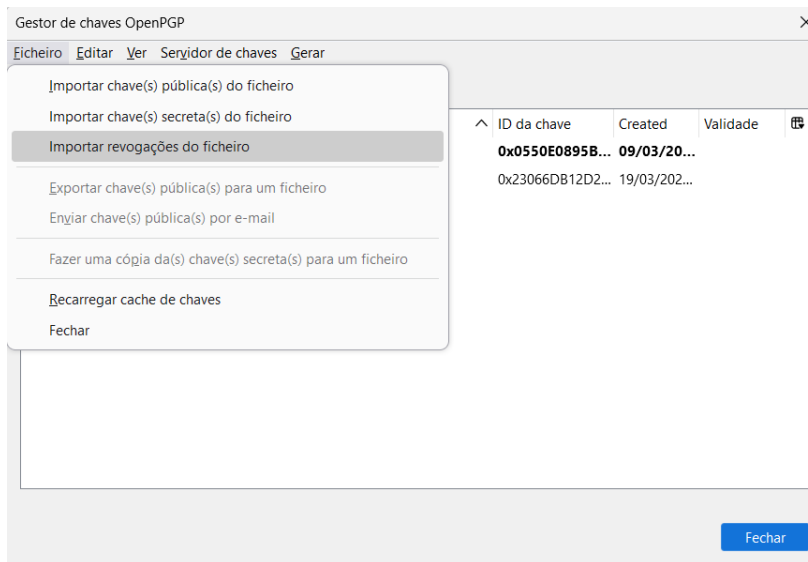


Figura 37: Importação do Certificado de Revogação.



Figura 38: Chave revogada no Thunderbird.

O cenário **pressupõe que a revogação seja publicada no servidor PGP** para que qualquer pessoa possa ter acesso a essa informação. Outra opção é a Inês enviar por email o certificado de revogação aos seus contactos, permitindo-lhes revogar a sua chave na lista de chaves PGP e evitar a sua utilização.

No entanto, se o Leonardo não estiver ciente da revogação da chave da Inês, ele pode inadvertidamente enviar um email encriptado com essa chave. Surpreendentemente, ele consegue enviar o email encriptado e a Inês consegue recebê-lo e lê-lo, conforme demonstrado na figura 39.

Nesta altura o emissor não conhece a data de revogação, pelo que, localmente, não há razão para não enviar. Só depois de fazer o

3.2. ENVIAR E RECEBER MENSAGENS SEGURAS

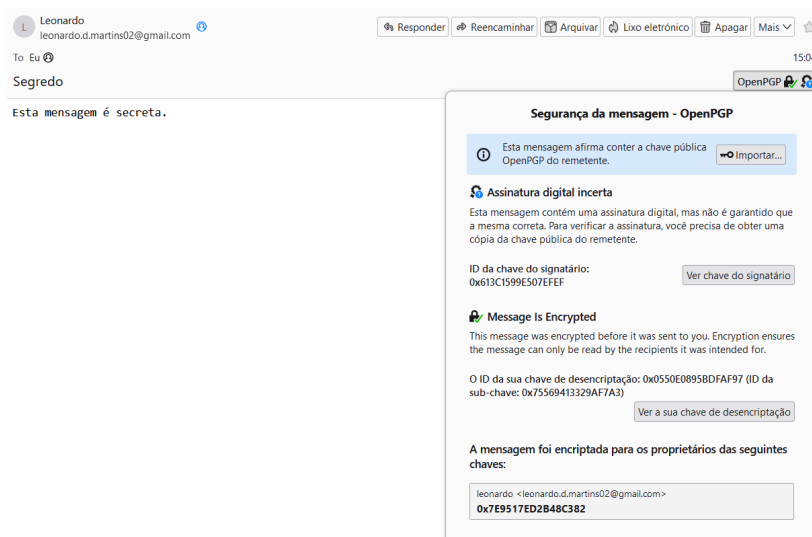


Figura 39: Receção do email encriptado do Leonardo para a Inês.

Se a Inês, ao enviar um email, anexar a sua chave pública, como esta foi revogada, o Thunderbird enviará automaticamente o certificado de revogação. Assim, o Leonardo pode revogar a chave da Eva no seu gestor de chaves OpenPGP. Na figura 47, observa-se o email que a Inês enviou ao Leonardo com o certificado de revogação em anexo.

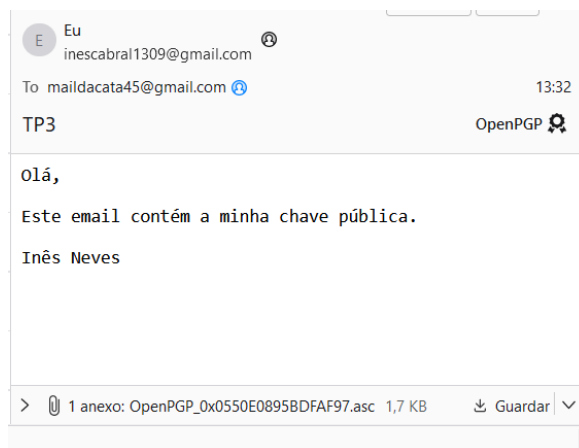


Figura 40: Email com certificado de revogação em anexo.

3.3. PROTEGER DOCUMENTOS LOCAIS

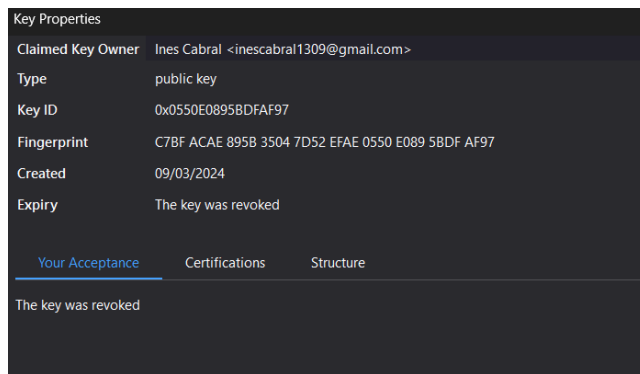


Figura 41: Chave revogada no gestor de chaves do Leonardo.

Como se pode observar na figura 42, o Leonardo tenta enviar um email para a Inês, mas não consegue porque a chave da Inês está revogada.

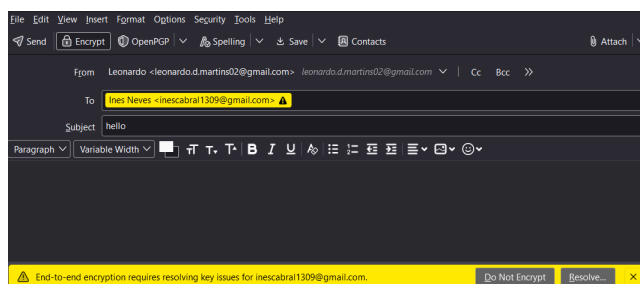


Figura 42: Tentativa de enviar email para a Inês.

3.3 Proteger documentos locais

De forma a proteger documentos locais o Kleopatra possui uma funcionalidade de assinar/cifrar pastas e ficheiros. Para tal fizemos as seguintes etapas. Começamos por selecionar a opção para cifrar a pasta dentro do menu do Kleopatra como pode ser observado na figura 43. Posteriormente, selecionamos o certificado e a pasta desejada, como demonstrado na figura 44. Por fim atingimos o objetivo final como apresenta a figura 45.

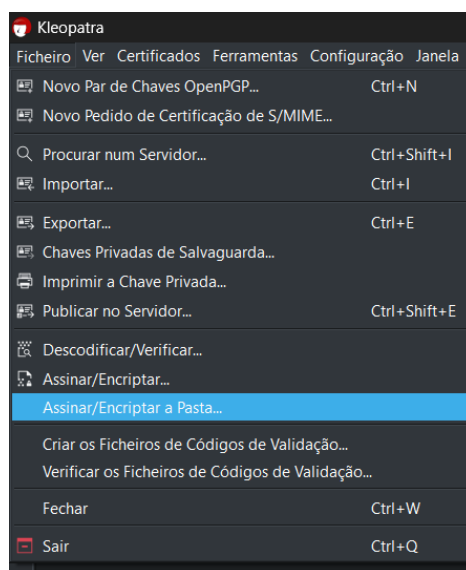


Figura 43: Opção para assinar/cifrar a pasta.

3.3. PROTEGER DOCUMENTOS LOCAIS

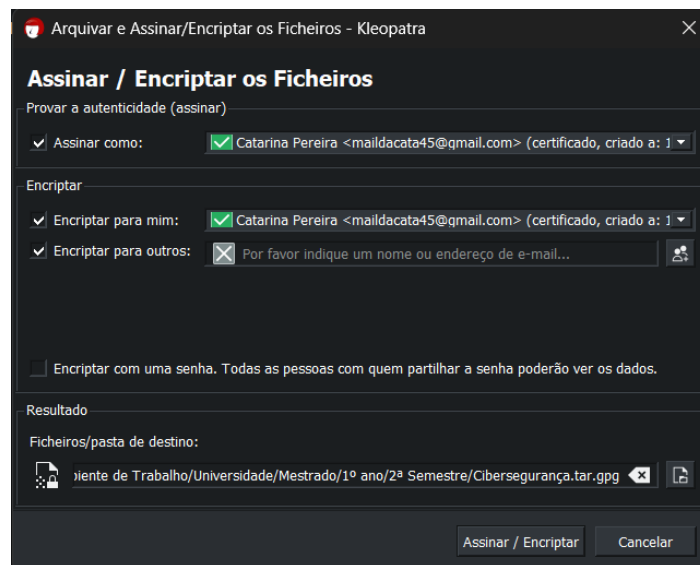


Figura 44: Escolha do certificado e pasta.

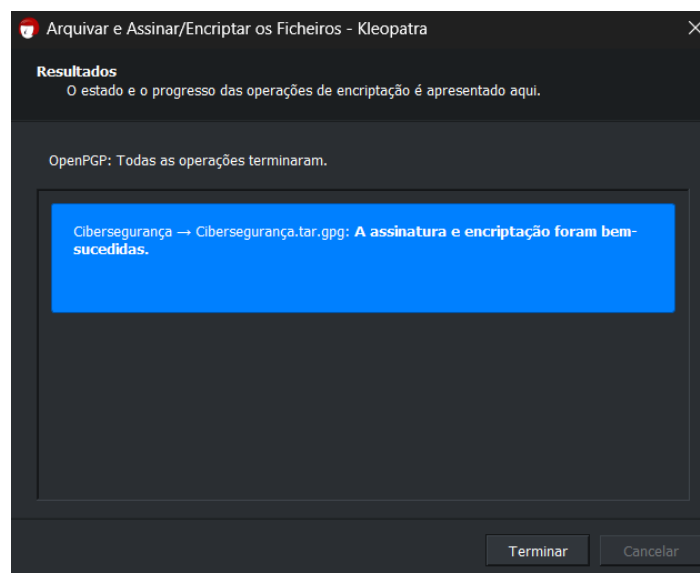


Figura 45: Confirmação da cifra.

4 Opção X509

Para operar com o X509, foi utilizado o OpenSSL na plataforma do sistema operacional Ubuntu. Considerando que o sistema operacional em uso já inclui o OpenSSL por padrão, basta verificar a versão instalada, Figura 46.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ openssl version
OpenSSL 3.0.2 15 Mar 2022 (Library: OpenSSL 3.0.2 15 Mar 2022)
```

Figura 46: Verificação da Versão do OpenSSL.

4.1 Gestão de chaves

Na Figura 47 e na Figura 48, observa-se um novo par de chaves, **do qual irá criar uma chave privada** e a chave pública associada, de 2048 bits, ambas guardadas no mesmo ficheiro, do tipo Privacy Enhanced Mail (PEM). Desta forma, a chave obtida é adequada para poder cifrar e assinar e não necessita de uma password para ser aplicada.

Um **arquivo PEM é um tipo de certificado digital** usado para trocar informações com segurança pela Internet [6]. É frequentemente usado na criptografia de e-mail e também pode ser usado para armazenar chaves privadas e certificados [7]. O arquivo é codificado em formato binário ou Base64 e normalmente possui uma extensão .pem [8].

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ openssl genrsa -out privkey.pem 2048
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ ls
privkey.pem
```

Figura 47: Criação de um novo par de chaves.

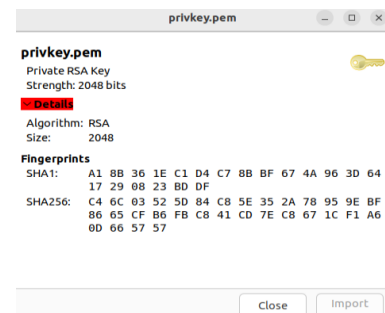


Figura 48: Verificação das chaves criadas.

Ao executar o comando de verificação da chave RSA, pode-se confirmar que a mesma foi gerada corretamente. Para assegurar a integridade da chave privada recém-criada, é recomendado utilizar o comando `openssl rsa -in privkey.pem -check`. Ao realizar essa verificação, obtêm-se a confirmação do bom estado da chave privada, como ilustrado na Figura 49.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ openssl rsa -in privkey.pem -check
RSA key ok
writing RSA key
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSIAgEAAQIBAQCI6v6ReF8nLaav
Ex/rD4aFFJ3eNp/geI5yvoT4ZNIrI19ys4A0u8f0eHY21b2Sc0GWSqTK2025U0d
8XzPD8HBAyoxo9Ctcfq2RHDQ042Mhs141tLW0o31X09+qPhs02VpZhuw91dV
HFGLT1w2MH8Q0sn1D/XBqu/r/CSHCz3sEd7Hsh+HfpoFj1VAK+481K/LGmuF+pBT
OpScMuYv6sTcdSEbn/ByuLVFc2dzthH3+l0k5KEj2X9zt660IXZltX2Q0XvtEae
t9L+WehBAon9H4+rdXGa8NHslpZhmessln25TgKX7LVBKXXVhNF0fx5MwHCNH2U
FhNjdfxAgMBAAECggEAAIzYBypuY97Vkd0CDVYHEKjfoZwGaYoGL2uyCSmD
qIegES390B0tlo0dpS+a5zh4S0VZsaa177UeV82HA/vAMZu2jG8gQWQvesqUZJAC
a/q6nXUfHUPwAsG12kIusbtZKBppq57nzE6pLM8kwcKfMBWbZrTpKd/UauLF0G
IC16k1fsjZB8PLNzr2H4dV0sD76s5Z5S12Blnltmb1a0o0GASCKWNNHf68r
e3nKz/DXRGHPZ2ML221k1ts+K38c5KT9RR1y53wF7DRF7ZouVeo5LPpkfInn1KA
l958e/cteznzLAe60ktLHAQn25gjd985wA19dh5QK8GQC+j6NB5E0zscpD11h
g7PIuLuy/ZMorkY/lo4X2zYAMB+LGIWUUG+sUjK9TnByXXCuaXZsfbwOCwLHsYB
BFYzW15luU2GSBzURDQmgDMD9dMvroEG+3rTobQYfYq4vgpea+l05r15gnYxa0e
873RI2awnjLhTtQo4R0WHF0rkwBgQC374y99PPQwznrdL6zYQY/+qV6H+Iw9G4e
kq95Z8THGAU2wBF0Fh4s4tmGKUCUdyLjeyLsoBQjd065208qbkhQRYbI6n1yUwn7
p777Xa2ZMf1uNEGy0nc/teshPEnHl0HLYFmpJyJwKMeGBckS+K6I+v7Gow
DCIVLd3LWKBgH0+Tszw2L13xIa1VERo4HkAKzFVA17g2OKLq/Kx8vqa4s5w9/H
02xSLKceASkTsQpFDtoT3yqxPa0u7PCZJecfrBKCnQl/H5NhzCmszt99t142weJ5
y1rM5tSAfoM9MJMVPGEUHFYH0gB+QD5zus+c1sE7bJ8T4C1MPhx18eLa0GAFhHD
foNBBCp5j5FoYuxSc1GhT9P16aXvQC7/Jgn8YzZGLFHNpDeV9eYtJtCF0Bhtz8
L18n/nGB7qeZtFN4T08cs+9jfyNLxziyL9lqkanS31BdLujzF06Gu0lnHLJ
68YDUQ25H1DT75GGeE0S0orLKW9jw95E94LMcgYA1ON9Ucknes6Pjy2I00+
+J5g8vduP154rdtI9pnpCYrN7zV19dyngLk4KoLYL3Vra04Jh+dk6ntWhavL
oL1tsGKJUE01U2AShQD1Cxy4mGn+0Df65w04GHquewVn110F2eyJA03pnh2Lz53Q
9seAY+7q+mWA9g+/9mFXKg==
-----END PRIVATE KEY-----
```

Figura 49: Verificação do estado da chave privada.

4.1. GESTÃO DE CHAVES

Para solicitar um certificado, é necessário preparar um arquivo contendo informações como a chave pública, dados pessoais e organizacionais, como o Common Name (CN) e o endereço de e-mail associados à chave privada previamente gerada. Este arquivo, conhecido como pedido de certificado, será encaminhado para a CA. Após a validação da identidade pela CA, o certificado assinado será emitido.

Após a elaboração do pedido de certificado com o comando `openssl req -new -key privkey.pem -out cert.csr`, procede-se à verificação do seu estado para garantir a sua integridade como representado na Figura 50.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ openssl req -new -key privkey.pem -out cert.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:PT
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:Guimaraes
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UMinho
Organizational Unit Name (eg, section) []:Dept DSI
Common Name (e.g. server FQDN or YOUR name) []:Grp5
Email Address []:pg53733@alunos.uminho.pt

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:cibergrupo5
An optional company name []:
```

Figura 50: Criação do certificado.

Depois do passo anterior, verificar-se-á o estado do ficheiro (de pedido de certificado) com o comando `openssl req -text -noout -verify -in cert.csr`, Figura 51.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ openssl req -text -noout -verify -in cert.csr
Certificate request self-signature verify OK
Certificate Request:
Data:
  Version: 1 (0x0)
  Subject: C = PT, ST = Some-State, L = Guimaraes, O = UMinho, OU = Dept DSI, CN = Grp5, emailAddress = pg53733@alunos.uminho.pt
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
      Public-Key: (2048 bit)
      Modulus:
        00:88:ea:fe:91:78:5f:27:2d:a6:af:13:1f:eb:0f:
        86:85:14:9d:de:34:6a:7f:81:e2:39:ca:fa:13:e1:
        93:48:44:8d:7d:ca:ce:00:0e:ef:2b:d1:e1:d8:db:
        56:f6:49:cd:06:59:2a:93:2b:63:b6:49:4d:1d:f0:
        1c:cf:0f:c1:c1:03:2a:31:a3:d0:42:b5:fa:89:cd:
        11:ce:40:ee:36:30:cb:35:e0:bb:64:58:3a:28:de:
        25:ce:f7:ea:85:5a:c9:0e:d9:5a:73:5a:ec:2b:f6:
        57:55:84:51:a2:4f:5c:36:58:7f:10:82:c9:f5:0f:
        fc:41:aa:e4:ff:ac:24:87:0b:3d:ec:11:de:cc:b0:
        7f:87:16:9a:05:8f:5b:c0:2b:ee:3c:d4:af:e5:1a:
        6b:85:fa:90:53:3a:9b:1c:5a:e6:2f:ea:c4:c2:77:
        91:1b:9b:fd:32:52:55:45:73:67:73:b6:11:c9:fa:
        2d:24:e4:a1:23:d9:7f:73:c6:de:ba:38:85:d9:8a:
        d5:f6:40:e5:ef:b4:46:9e:b7:d2:fe:59:e8:7c:02:
        89:fd:1f:8f:ab:75:71:9a:f0:d1:ec:8a:96:61:99:
        eb:2c:96:6d:92:4e:02:97:ee:2b:fc:29:75:d8:84:
        d1:45:39:fc:52:33:01:dc:34:76:54:16:19:cd:8d:
        d7:f1
      Exponent: 65537 (0x10001)
  Attributes:
    challengePassword :cibergrupo5
    Requested Extensions:
  Signature Algorithm: sha256WithRSAEncryption
  Signature Value:
    1e:a6:49:c1:d2:93:8d:34:d8:9c:f2:6b:b3:c9:ef:ec:58:4b:
    6e:cf:bd:f1:88:6d:9a:eb:0b:c2:1d:6b:a8:66:19:e0:72:76:
    85:fe:6f:d4:4b:d7:1f:50:8b:b1:6b:07:9d:55:e8:bd:c7:de:
    dd:79:d0:aa:f4:3c:00:3d:7f:c7:e6:20:4d:7e:4b:e8:18:72:
    16:dd:f2:9a:77:d1:ae:3f:19:3b:d3:4d:06:08:6c:1d:a9:a8:
    01:75:4b:43:66:78:61:77:ba:31:78:1f:25:be:87:b2:7e:5b:
    86:e4:3d:f3:29:28:68:c2:19:58:8e:47:00:e1:a9:a8:04:b0:
    9e:1d:6c:b4:49:be:31:8c:65:72:f9:c1:20:a5:83:ca:2a:c1:
    b5:10:be:0e:ad:74:0b:fa:27:9d:81:b2:a5:4f:af:de:4b:c1:
    e5:7f:01:40:cf:68:45:6b:b5:3d:76:8f:39:f2:dc:5d:e4:67:
    15:7f:b7:ee:66:fe:89:d5:ab:6f:f4:13:b4:8a:50:11:a9:40:
    78:1f:8f:1d:06:41:59:d7:a6:ab:c0:20:93:b9:e5:28:91:64:
    ff:ef:cb:3d:76:61:b6:c1:ab:e0:88:bc:69:d5:9a:a7:55:b8:
    18:e3:37:c6:d3:00:18:dd:5f:46:6d:8f:32:db:0d:ea:8c:3e:
    02:e1:de:a0
```

Figura 51: Verificação do estado de pedido de certificado.

Neste ponto de situação, foi gerado um certificado auto-assinado utilizando a chave privada. A utilização de certificados auto-assinados é útil para importar a chave privada em cenários específicos. Para gerar o certificado auto-assinado, utiliza-se o comando `openssl x509 -req -in cert.csr -signkey privkey.pem -out privcert.crt`.

O comando exemplificado na Figura 52 especifica que o certificado será do tipo x509, assinado pela chave

4.1. GESTÃO DE CHAVES

privada com o nome indicado. Após a geração do certificado auto-assinado com a chave privada previamente criada, verificou-se que o mesmo é válido e contém os valores previamente inseridos.

```
Catarina@catarina-VirtualBox: ~/ciberseguranca/TP3$ openssl x509 -req -in cert.csr -signkey privkey.pem -out privcert.crt
Certificate request self-signature ok
subject=C = PT, ST = Some-State, L = Guimaraes, O = UMinho, OU = Dept DSI, CN = Grp5, emailAddress = pg53733@alunos.uminho.pt
```

Figura 52: Criação do certificado auto-assinado.

Ao utilizar o comando `openssl x509 -text -in privcert.crt`, é possível validar o estado do arquivo que contém o certificado auto assinado e os elementos essenciais associados a ele. Este comando permite visualizar informações relevantes presentes no certificado, como a validade, a chave RSA e o algoritmo de assinatura.

Após verificar o estado do certificado, conforme ilustrado na Figura 53, são apresentados dados fundamentais utilizados no pedido de certificado, fornecendo detalhes como a validade, a chave RSA e o algoritmo de assinatura.

[illegible]

Figura 53: Criação do certificado auto-assinado.

Na etapa atual, o grupo decidiu criar um PKI simples [9]. Começou a ser criado um repositório local, conforme mostrado na Figura 54.

4.1. GESTÃO DE CHAVES

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ git clone https://bitbucket.org/stefanholek/pki-example-1
Cloning into 'pki-example-1'...
Unpacking objects: 100% (79/79), 8.36 KiB | 372.00 KiB/s, done.
catarina@catarina-VirtualBox:~/ciberseguranca/TP3$ cd pki-example-1
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ mkdir -p ca/root-ca/private ca/root-ca/db crl certs
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$
```

Figura 54: Criação de um repositório remoto.

Após o descarregamento do repositório mencionado, foram estabelecidas uma série de diretorias destinadas a armazenar os certificados, conforme ilustrado na Figura 55.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ mkdir -p ca/signing-ca/private ca/signing-ca/db crl certs
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ chmod 700 ca/signing-ca/private
```

Figura 55: Criação da diretorias.

Para organizar informações de maneira eficiente, é comum recorrer à utilização de bases de dados. Estas podem ser visualmente representadas, como exemplificado na Figura 56.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ mkdir -p ca/signing-ca/private ca/signing-ca/db crl certs
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ chmod 700 ca/signing-ca/private
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ cp /dev/null ca/signing-ca/db/signing-ca.db
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ cp /dev/null ca/signing-ca/db/signing-ca.db.attr
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ echo 01 > ca/signing-ca/db/signing-ca.crt.srl
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ echo 01 > ca/signing-ca/db/signing-ca.crl.srl
```

Figura 56: Criação da base de dados.

Durante esta etapa, é essencial solicitar a autenticação do certificado e, em seguida, criar uma chave privada para a autoridade certificadora raiz (root CA).

O comando `openssl req -new` é utilizado para solicitar a autenticação mencionada, cujo resultado pode ser visualizado na Figura 57.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ openssl req -new -config etc/signing-ca.conf -out ca/signing-ca.csr -keyout ca/signing-ca/private/signing-ca.key
.....
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
.....
```

Figura 57: Pedido de autenticação de certificado.

Na Figura 58, como se pode ver foi criado o certificado CA auto assinado.

```
catarina@catarina-VirtualBox:~/ciberseguranca/TP3/pki-example-1$ openssl ca -selfsign -config etc/root-ca.conf -in ca/root-ca.csr -out ca/root-ca.crt -extensions root_ca_ext
Using configuration from etc/root-ca.conf
Enter pass phrase for ./ca/root-ca/private/root-ca.key:
40D7840E137F000:error:0700000c:configuration file routines:NCONF_get_string:no value:./crypto/conf/conf_lib.c:315:group=<NULL> name=unique_subject
Check that the request matches the signature
Signature ok
Certificate Details:
  Serial Number: 1 (0x1)
  Validity
    Not Before: Mar 23 17:06:32 2024 GMT
    Not After : Mar 23 17:06:32 2034 GMT
  Subject:
    domainComponent           = uminho
    domainComponent           = aluno
    organizationName          = Universidade do Minho
    organizationalUnitName    = CiberseguranE7a
    commonName                 = Simple Root CA
  X509v3 extensions:
    X509v3 Key Usage: critical
      Certificate Sign, CRL Sign
    X509v3 Basic Constraints: critical
      CA:TRUE
    X509v3 Subject Key Identifier:
      76:F5:2F:94:E1:07:87:77:8C:CA:CB:69:15:96:A0:46:7B:42:BB:1A
    X509v3 Authority Key Identifier:
      76:F5:2F:94:E1:07:87:77:8C:CA:CB:69:15:96:A0:46:7B:42:BB:1A
Certificate is to be certified until Mar 23 17:06:32 2034 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]:y
Write out database with 1 new entries
Data Base Updated
```

Figura 58: Criação de um certificado CA auto assinado.

O passo seguinte foi criar a Signing Ca, como se pode observar na Figura 59, e para isso foram usados os mesmos comandos usados no Root CA, para criar a base de dados e o pedido e certificado da Signing CA.

4.1. GESTÃO DE CHAVES

```

$ cd /etc/ssl/certs/; openssl ca -config etc/root-ca.conf -in ca/signed-ca.csr -out ca/signed-ca.crt -extensions signing_ca_ext
Using configuration from etc/root-ca.conf
Enter pass phrase for ./ca/root-ca/private/root-ca.key:
check that the request matches the signature
Signature ok
Certificate Details:
  Serial Number: 2 (0x2)
  Validity
    Not Before: Mar 23 17:12:02 2024 GMT
    Not After : Mar 23 17:12:02 2034 GMT
  Subject:
    domainComponent           = uminho
    domainComponent           = aluno
    organizationName          = Universidade do Minho
    organizationalUnitName    = CibersegurançaFEA
    commonName                 = Simple Signing CA
  X509v3 extensions:
    X509v3 Key Usage: critical
      Certificate Sign, CRL Sign
    X509v3 Basic Constraints: critical
      CA:TRUE, pathlen:0
    X509v3 Subject Key Identifier:
      BC:59:C6:06:71:2A:57:95:P9:00:7A:29:2D:DF:8C:27:44:68:9D:B0
    X509v3 Authority Key Identifier:
      76:FS:2F:94:E1:07:87:77:8C:CA:CB:69:15:96:A0:46:7B:42:BB:1A
Certificate is to be certified until Mar 23 17:12:02 2034 GMT (3652 days)
Sign the certificate [y/n]y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated

```

Figura 59: Criação do certificado CA certificado pela Root CA

Na Figura 60, utilizando o Signing CA, foi criado o pedido de certificado de email.

```

root@kali:~# openssl req -new -config etc/email.conf -out certs/catarina.csr -keyout certs/catarina.key
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

1. Domain Component           (eg, com)      []:uninho
2. Domain Component           (eg, company) []:uninho
3. Domain Component            (eg, pklt)    []:aluno
4. Organization Name           (eg, company) []:Universidade do Minho
5. Organizational Unit Name     (eg, section) []:Ciberseguranca
6. Common Name                 (eg, full name) []:Catarina Pereira
                                (eg, email address) []:paola@catas@opmail.com

```

Figura 60: Criação de um pedido de certificado de email.

Pode-se observar na Figura 61 que a Signing CA aceitou o pedido de certificado e por isso o certificado foi criado.

```

cat@macatatarina-VirtualBox: /Users/roger/Work/TP3/pkt-example: $ openssl ca -config etc/signing-ca.conf -in certs/catarina.csr -out certs/catarina.crt -extensions email_ext
Using configuration from etc/signing-ca.conf
Enter pass phrase for ./ca/signing-ca/private/signing-ca.key:
40B7BC3916730B00:error:0700006C:configuration file routines:NCONF_get_string:no value:../crypto/conf/conf_ltb.c:315:group=<NULL> name=unqute_subject
check that the request matches the signature
Signature ok
Certificate Details:
  Serial Number: 1 (0x1)
  Validity
    Not Before: Mar 23 17:14:08 2024 GMT
    Not After : Mar 23 17:14:08 2026 GMT
  Subject:
    domainComponent           = uminho
    domainComponent           = uminho
    domainComponent           = aluno
    organizationName          = Universidade do Minho
    organizationalUnitName    = CiberseguranE7a
    commonName                 = Catarina
X509v3 extensions:
  X509v3 Key Usage: critical
    Digital Signature, Key Encipherment
  X509v3 Basic Constraints:
    CA:FALSE
  X509v3 Extended Key Usage:
    E-mail Protection, TLS Web Client Authentication
  X509v3 Subject Key Identifier:
    ID:4B:FC:22:74:5B:F3:EE:C4:8D:69:93:9A:30:8B:AA:B3:03:CE:90
  X509v3 Authority Key Identifier:
    BC:59:CE:D6:7E:2A:57:95:F9:00:7A:29:2D:0F:8C:27:44:68:9D:B0
  X509v3 Subject Alternative Name:
    email:ma1ldacata45@gmail.com
Certificate is to be certified until Mar 23 17:14:08 2026 GMT (730 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
write out database with 1 new entries
Data Base Updated.

```

Figura 61: Aceitação do pedido por parte da Signing CA e criação do certificado.

Para se poder adicionar o certificado ao Thunderbird é necessário um ficheiro no formato PKCS12. Para tal, teve-se de executar o comando presente na Figura 62.

Da Catarina. E dos restantes?!O que são os

4.1. GESTÃO DE CHAVES

```
catn@catarina-virtualbox: ~/catn$ openssl pkcs12 -export -in certs/catarina.crt -inkey certs/catarina.key -certfile ca/signing-ca.crt -name "Catrina Pereira" -out catarina.pvk12.p12
Enter pass phrase for certs/catarina.key:
Enter Export Password:
Verifying - Enter Export Password:
```

Figura 62: Criação do ficheiro PKCS12

Após a criação do ficheiro em formato PKCS12 foi verificado o estado do mesmo como podemos ver na Figura 63 e na Figura 64.

[illegible]

Figura 63: Verificação do ficheiro PKCS12 (1).

[illegible]

Figura 64: Verificação do ficheiro PKCS12 (2).

4.2 Enviar e receber mensagens seguras

De modo a se comunicar encriptadamente e de forma segura, é necessário a importação dos certificados a usar.

Para adicionar o certificado com sucesso, **é necessário utilizar a versão 102.8.0 do Thunderbird, que é a versão do ano passado.** Nas versões mais recentes, essa adição não é permitida, o que levou à instalação de uma versão mais antiga. O registo do insucesso ao adicionar o certificado com uma versão mais recente não foi feito. Na Figura 65, observa-se a importação do certificado do remetente.

Porquê? Co

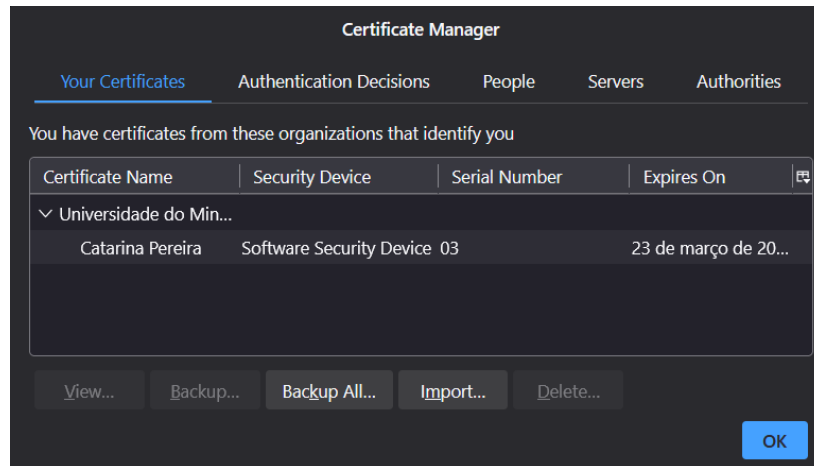


Figura 65: Import do certificado.

No caso da Figura 65, observa-se a importação do certificado da autoridade certificadora criada.

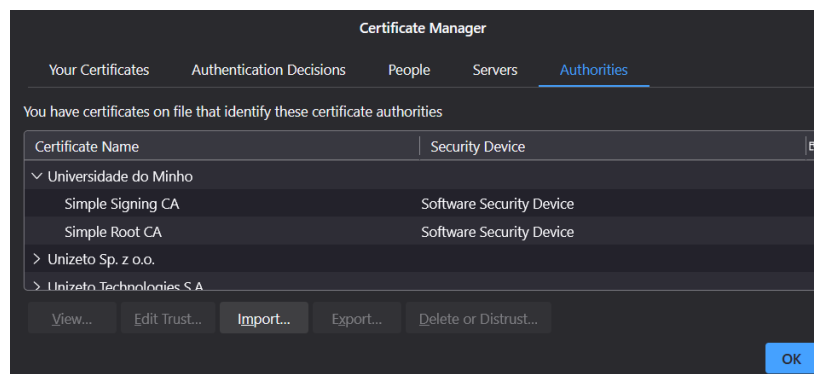


Figura 66: Import do certificado da autoridade certificadora criada.

Foi necessário escolher que certificados usar para assinar e decifrar as mensagens, tendo escolhido o mesmo para as duas funções, como é possível ver na Figura 69

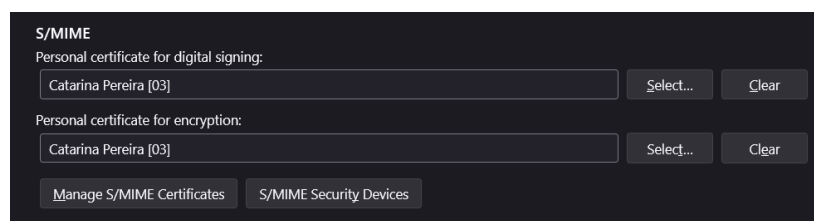


Figura 67: Seleção dos certificados para assinar e decifrar.

Com esta etapa concluída, **resta apenas a substituição das mensagens.** É importante notar que no recetor foi seguido o mesmo procedimento, **exceto no passo de importar o próprio certificado e no certificado do remetente,**

4.2. ENVIAR E RECEBER MENSAGENS SEGURAS

```
catarina@catarina-VirtualBox: ~/ciberseguranca/TP3/pki-example-1$ openssl ca -gencrl -config etc/signing-ca.conf -out crl/signing-ca.crl
Using configuration from etc/signing-ca.conf
Enter pass phrase for ./ca/signing-ca/private/signing-ca.key:
```

Figura 71: Obtenção da CRL.

Após a revogação do certificado, não houve nenhuma ocorrência. Tentou-se enviar o e-mail assinado, **sem sucesso**. O que ocorreu foi exatamente o mesmo que foi demonstrado antes da revogação do certificado.

Ao consultar a lista da Figura 72, foi constatado que os certificados selecionados estão incluídos nela. No entanto, devido à falta de suporte da CA implementada para a verificação de certificados revogados e apesar de várias tentativas, o Thunderbird permite a utilização desse certificado sem emitir qualquer aviso. Isto acarreta sérios problemas na integridade da segurança da comunicação. Para resolver essa questão, é crucial obter os certificados de uma CA legítima e confiável, que mantenha e forneça documentação atualizada sobre o status de validade dos certificados emitidos.

Ou era necessário garantir que o thunderbird

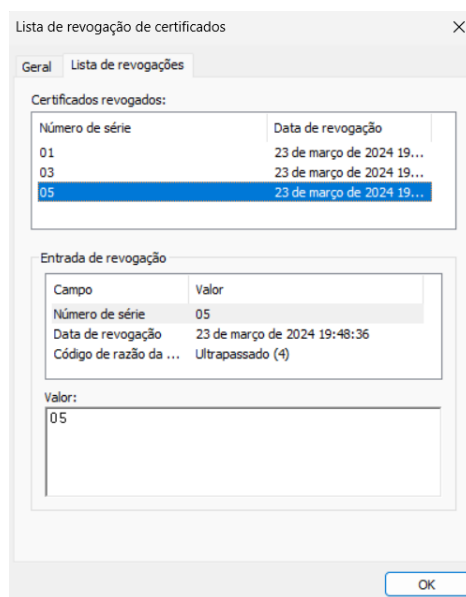


Figura 72: Lista de CRL.

Ou seja, desta forma, para garantir a validade do certificado, o próprio utilizador teria de verificar a CRL atualizada e discernir quais os certificados válidos.

A revogação através do mecanismo Online Certificate Status Protocol (OCSP) não foi realizada devido à falta de tempo. O objetivo era verificar se, por meio deste método, o grupo conseguiria realizar uma revogação bem-sucedida.

5 Conclusão

Com a conclusão deste trabalho prático, o grupo realizou todos os passos e correspondeu ao proposto do trabalho prático. Tendo também desenvolvido competências de trabalho com certificados PGP e X509.

No PGP, a adição de subchaves foi realizada já depois de o grupo ter terminado todos os passos, fazendo com que a subchave adicionada já fosse adicionada como revogada, mas ainda assim o grupo conseguiu adicionar a mesma.

Referências Bibliográficas

- [1] Sourabh Chandra et al. "A comparative survey of Symmetric and Asymmetric Key Cryptography". Em: IEEE, nov. de 2014, pp. 83–93. ISBN: 978-1-4799-5748-4. DOI: 10.1109/ICECCE.2014.7086640.
- [2] José Carlos Bacelar Ferreira Junqueira Almeida. *Criptografia Assimétrica Criptografia e Segurança de Redes*. Acedido a 09 de março de 2024. Dezembro de 2022.
- [3] José Carlos Bacelar Ferreira Junqueira Almeida. *Certificados Digitais Criptografia e Segurança de Redes (LEI)*. Acedido a 09 de março de 2024. Dezembro de 2022.
- [4] P. Wing e B. O'Higgins. "Using public-key infrastructures for security and risk management". Em: *IEEE Communications Magazine* 37 (9 1999), pp. 71–73. ISSN: 01636804. DOI: 10.1109/35.790867.
- [5] *Como gerar chaves PGP com GPG?* <https://pt.linux-console.net/?p=15179>. Acedido a 9 de março de 2024.
- [6] William Jones. "Personal Information Management". Em: *Annual Review of Information Science and Technology* 41 (1 jan. de 2007), pp. 453–504. ISSN: 0066-4200. DOI: 10.1002/aris.2007.1440410117.
- [7] Ofer Bergman et al. "Personal information management". Em: ACM, abr. de 2004, pp. 1598–1599. ISBN: 1581137036. DOI: 10.1145/985921.986164.
- [8] Haris Zafeiropoulos et al. "PEMA: from the raw .fastq files of 16S rRNA and COI marker genes to the (M)OTU-table, a thorough metabarcoding analysis". Em: (). DOI: 10.1101/709113. URL: <https://doi.org/10.1101/709113>.
- [9] *Simple PKI — OpenSSL PKI Tutorial*. Acedido a 22 de março de 2024. URL: <https://pki-tutorial.readthedocs.io/en/latest/simple/index.html>.

referências incor