Universidade da Beira Interior
Faculdade de Engenharia
Departamento de Informática

Segurança Informática
Aula 7
Licenciatura em Engenharia Informática
Licenciatura em Informática Web
Licenciatura em Tecnologias e Sistemas da Informação

Computer Security
Lecture 7
Degree in Computer Science and Engineering
Degree in Web Informatics
Degree in Information Technologies and Systems

Sumário
Análise de como a criptografia moderna se concretiza em ferramentas ou recursos que podem ser usados na prática. Discussão da ferramenta *Pretty Good Privacy* – PGP (Privacidade Razoavelmente Boa) e estudo do caso do cartão do cidadão da república portuguesa.

Summary
Analysis of how modern criptography results in tools or resources that can be used in practice. Discussion of the Pretty Good Privacy (PGP) tool and case study of the portuguese citizen card.

1 Criptografia ao Nível da Camada de Aplicação: Privacidade Rasoavelmente Boa

Cryptography at the Application Layer: Pretty Good Privacy (PGP)

1.1 Introdução e História
Introduction and History

O acrónimo PGP abrevia a expressão inglesa ***Pretty Good Privacy*** e é utilizado para designar o ***software*** **que combina** diversos mecanismos da **criptografia de chave simétrica e de chave pública** reconhecidamente fortes para **fornecer serviços de segurança** a comunicações eletrónicas, nomeadamente **e-mail**, e armazenamento de dados. Atualmente, a versão aberta do PGP é **especificada no** *Request for Comments* **(RFC) 4880**, intitulado *OpenPGP Message Format*, e datado de 2007[1].

O PGP é atualizado regularmente, de modo a integrar sempre os mecanismos do estado da arte nesta área do conhecimento. Proporciona serviços de confidencialidade, autenticação do emissor de mensagens e integridade dos dados. Foi desenvolvido originalmente por **Phil Zimmerman** (um **então aluno da Universidade** de Atlantic Florida, USA, em 1991). Depois da sua incarnação como OpenPGP, a combinação na forma de *software* tornou-se uma norma livre e aberta:
"Originally designed as a human rights tool, PGP was published for free on the Internet in 1991." in `http://www.philzimmermann.com/EN/background/index.html`.

Apesar do PGP poder ser usado para proteger dados em disco a longo-prazo, é **usado principalmente para assegurar a privacidade em troca de correio eletrónico**. Há, por isso, **várias aplicações que implementam o PGP** (e.g., o **GPG** – *GNU Privacy Guard*, em Linux), e é possível encontrar vários *plugins* para os clientes de *e-mail* mais conhecidos, e.g., *Outlook*, *Thunderbird*, etc.

O PGP constitui um esforço muito bem conseguido de **tornar a criptografia disponível ao utilizador**, requerendo pouco esforço, para além do inicial, para que este usufrua dos seus serviços. Algumas distribuições de Linux (e.g., Arch Linux ou Fedora) **recorrem ao PGP** para garantir que os pacotes de *software* que são instalados via gestor de pacotes chegam íntegros ao sistema e provêm de fonte fidedigna.

[1] Ver, e.g., `http://tools.ietf.org/html/rfc4880`.

1.2 Serviços Disponibilizados pelo PGP
Services Provided by PGP

O PGP define e **oferece essencialmente 5 serviços** diferentes para manuseamento e transmissão de ficheiros (a mensagem de *e-mail* é tratada como um ficheiro):

Autenticação – discutida em baixo;

Confidencialidade – também discutida em baixo;

Compressão – também mencionada a seguir;

Serviço de compatibilidade – o ficheiro cifrado é codificado em caracteres ASCII usando codificação Base64, garantindo que qualquer mensagem é guardada ou transmitida numa forma que mais dificilmente é erradamente interpretada depois de devolvida ou transmitida;

Serviço de Segmentação – se o ficheiro for grande, o PGP permite dividi-lo e cifrar os vários segmentos.

1.3 Gestão de Chaves no PGP
Key Management in PGP

Quando um utilizador **instala e inicializa** uma aplicação que implemente o PGP, **tem de** necessariamente **gerar pares de chaves públicas e privadas ou importar** pares de chaves que já possua. Como ficará claro adiante, estas chaves são centrais ao funcionamento deste *software*. Como normalmente são usadas com **dois objetivos diferentes** (**confidencialidade** e **autenticação** ou assinatura digital), o utilizador começa quase **sempre por criar (ou importar) 2 pares** de chaves públicas ou privadas: um par só para assinaturas digitais e outro par só para confidencialidade. A utilização destas chaves pode ser resumida da seguinte forma:

1. O utilizador usa a **sua chave privada**, do par definido para **autenticação**, para fazer a sua **assinatura digital** nas mensagens que envia;
2. O utilizador usa a **chave pública do recipiente**, do par definido para **autenticação**, para fazer a **verificação** de assinaturas recebidas deste;
3. O utilizador usa a **chave pública do recipiente**, do par definido para **confidencialidade**, para **cifrar** chaves de sessão;
4. O utilizador usa a sua chave privada, do par definido para **confidencialidade**, para **decifrar** chaves de sessão.

Numa implementação PGP, as chaves assimétricas são **guardadas em dois chaveiros** diferentes:

1. Um **chaveiro privado**, que contem **uma ou mais chaves privadas** do utilizador. Este chaveiro está **protegido por uma palavra ou frase-passe**.
2. Um **chaveiro público**, que o utilizador vai **manualmente ou semi-automaticamente povoando com chaves públicas de outros utilizadores**.

Em baixo mostram-se exemplos (simplificados e adaptados) de um chaveiro privado (primeira tabela) e um público (segunda tabela):

UserID	KeyID	Public Key	Encrypted Priv. Key	Timestamp
kak@abc.com	EA132...43	EA132...43...A21	34ABF23...A9	041908–11:30
kak@abc.com	...	...	...	...

UserID	KeyID	Public Key	Producer Trust	Certificate	Certificate Trust	Key Legitimacy	Timestamp
kak@abc.com	EA132...43	EA132...43...A21	Full	—	—	Full	041908–11:30
zaza@foo.com	132AB...02	132AB...02...23A	Full	—	—	Full	050208–13:25
toto@bar.com	231DA—02	231DA—02...33E	Full	Zaza's	Full	Full	050308–12:01
...	...	...	...	...	...	...	

Como cada utilizador pode ter **mais do que um par de chaves**, este informa o recipiente de determinada mensagem acerca do par utilizado **através do seu identificador** (ID), que é normalmente composto pelos **primeiros bytes da respetiva chave pública**. A combinação do **ID do utilizador** (tipicamente o endereço de ***e-mail***) com o **ID da chave** carateriza univocamente um par de chaves. Quando se envia uma mensagem, **envia-se também a informação de qual foi a chave utilizada**.

O **chaveiro privado** só contém chaves privadas e portanto **terá poucas entradas** (tantas quantas as chaves privadas de um utilizador). O **chaveiro público terá tantas entradas quantas as chaves importadas** pelo utilizador, **mais as públicas** que lhe pertencem também a ele(a). O chaveiro público **é preenchido através de** uma das formas seguintes:

1. Através de **troca direta**. Por exemplo, se determinado utilizador exporta a sua chave pública e cola a como assinatura nos seus *e-mails* ou o utilizador encontra-se com outro numa conferência e trocam chaves usando *pens*. A troca direta deve ser acompanhada de mecanismos que permitam comprovar que a chave é realmente daquele utilizador (e.g., usar o telefone) e está integra.
2. Através de **servidores** que publicam as chaves dos utilizadores, até que alguém as requisite. Estes servidores não são autoridades de certificação.

Note-se que o PGP **continua a sofrer do problema associado à criptografia de chave-pública**, nomeadamente relacionado com a dificuldade em garantir que uma dada chave pública pertence ou não a uma entidade específica. Anteriormente, foi discutido que **a *Public Key Infrastructure* (PKI) dava resposta a esse problema** usando uma estrutura hierárquica de Autoridades de Certificação em quem um utilizador confia. No caso d**o PGP, isto resolve-se recorrendo ao conceito de Rede de Confiança**.

1.4 Rede de Confiança

Web of Trust

A confiança de que determinada chave pertence a determinado utilizador constrói-se de maneira diferente em PGP, já que **não existe a estrutura hierárquica** (pelo menos diretamente) que existe para a PKI. Neste caso, o

conceito central à construção de confiança é o de Rede de Confiança (adaptação da designação inglesa *Web of Trust*) que se baseia levemente na hipótese de existirem 6 graus de separação entre quaisquer 2 seres humanos no planeta (em média). Esta forma de estabelecer confiança baseia-se n**um sistema completamente descentralizado** e não necessita da definição ou implementação de entidades terceiras em quem temos de confiar. Em vez disso, só precisamos de **estabelecer uma política de confiança em relação a chaves que possamos importar e estabelecer** – pessoal e por vezes manualmente – **confiança noutros utilizadores do PGP**.

O **primeiro mecanismo** desta *Web of Trust* consiste em **assinar digitalmente as chaves em que confiamos** (i.e., as chaves que sabemos pertencerem aos utilizadores a que estão associadas). **Caso tenhamos a certeza da correção de determinada chave que importamos** para o chaveiro, **assinamo-la com a nossa chave privada**. **Este procedimento diz ao nosso *software* que temos confiança absoluta nessa chave.**

Caso queiramos atestar a validade de uma chave e **partilhar isso com outros**, **assinamo-la e exportamos esse certificado/assinatura** juntamente com a chave. No PGP é, pois, normal receber chaves públicas juntamente com assinaturas digitais feitas por outros utilizadores. Elaborando uma política de confiança relativamente a outros utilizadores, pode-se então determinar se uma chave está correta e é de confiança ou não. Para isso, usam-se **três níveis de confiança relativamente a outros utilizadores**:

1. Confiança **Absoluta** – que define que determinado utilizador é **um *trusted introducer* e que confiamos cegamente em todas as chaves que sejam endossadas por ele(a)**, i.e., confiamos em todas as chaves que vêm com uma assinatura feita por ele(a).
2. Confiança **Marginal** – que define que não confiamos totalmente nesse utilizador, mas que **podemos eventualmente vir a confiar na chave caso vários utilizadores com confiança marginal a assinem**. Por defeito, algumas implementações ajustam para 3 o número de assinaturas relativas a utilizadores com confiança marginal necessário para que se confie em determinada chave.
3. **Sem Confiança** – que determina que não confiamos em chaves que tragam a assinatura deste utilizador.

De um modo resumido, pode-se dizer que a *Web of Trust* determina o seguinte:

1. **Confiamos em todas as chaves assinadas por nós**.
2. Confiamos em todas as chaves assinadas por uma chave em que confiamos totalmente (nível 1).
3. Confiamos nas chaves que foram assinadas por, pelo menos, n chaves em que confiamos marginalmente (nível 2). Este n pode ser ajustado pelo utilizador, mas é normalmente 3.

1.5 Revogação de Chaves
Key Revocation

Havendo chaves públicas, também existe a necessidade de definir mecanismos para as inutilizar em caso destas serem **comprometidas** ou **deixarem de ser seguras**. No PGP, apontam-se duas formas para conseguir isso:

1. **O utilizador emite um certificado a revogar a sua própria chave pública**, assinando-a com a sua chave privada. Este mecanismo não pode ser usado no caso em que o utilizador está a revogar a chave porque as perdeu.
2. Um utilizador **nomeia outro utilizador** (em quem tem absoluta confiança) para que este possa revogar as suas chaves caso as tenha perdido (por exemplo). Neste caso, o utilizador emite um documento, assinado por ele, onde atesta isto, **quando ainda tem as chaves**. Caso venha a precisar de revogar as chaves (e.g., porque as perdeu), pede ao outro que emita um certificado de revogação, juntando-o ao documento antes mencionado.

1.6 Serviço de Autenticação (Assinatura Digital)
Authentication Service (Digital Signature)

O serviço de autenticação em PGP é sinónimo de assinatura digital. Significa que, no contexto de uma comunicação eletrónica, a Alice assegura ao Bob que foi ela quem escreveu determinada mensagem. Em baixo, inclui-se um exemplo de um *e-mail* com uma mensagem completamente fictícia, só para ilustrar a explicação que se segue:

```
--BEGIN PGP SIGNED MESSAGE--
Hash:  SHA1

Bob,My husband is out of town
tonight.Passionately yours, Alice

--BEGIN PGP SIGNATURE--
Version:  PGP 5.0
Charset:  noconv
yhHJRHhGJGhgg/12EpJ+lo8gE4vB3mqJhFEvZP9t6n7G6m5Gw2
--END PGP SIGNATURE--
```

Como se pode ver, a mensagem é codificada em base 64 (só tem os caracteres [a–z], [A–Z] e [0..9]) para efeitos de compatibilidade.

O protocolo envolvido no envio de uma mensagem autenticada é o seguinte:

1. Alice (A): obtém **a sua chave privada** do par usado para assinatura digital **do chaveiro privado** (é necessário **inserir a sua palavra ou frase-chave**);

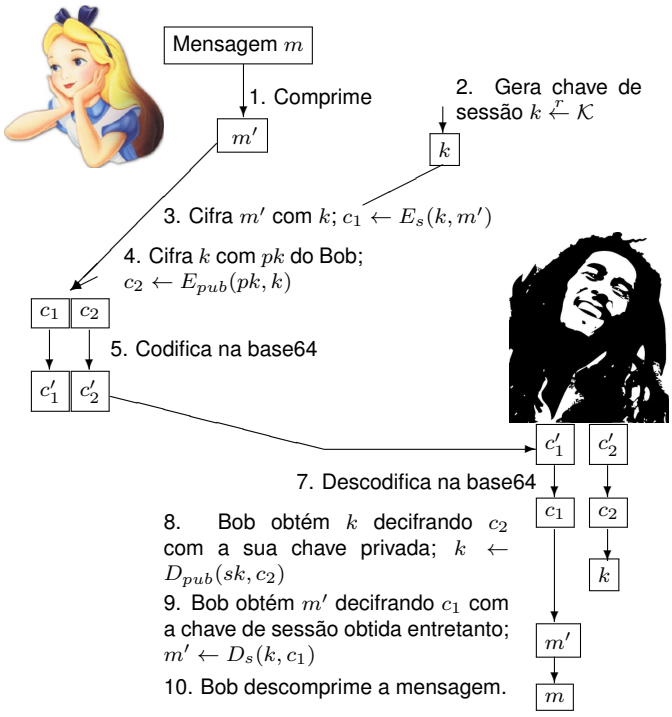
2. A: $t = S(sk, m)$
   A Alice produz a assinatura digital da mensagem m;
3. A: m_{64} = $BASE64(m)$ e t_{64} = $BASE64(t)$
   A **Alice codifica** a mensagem e a assinatura digital em Base 64;
4. A→ Bob (B): m_{64}, t_{64}
   A Alice envia a mensagem e a assinatura digital em Base64 ao Bob.
5. B: $m = BASE64^{-1}(m_{64})$ e $t = BASE64^{-1}(t_{64})$;
   Bob **descodifica a mensagem** e a assinatura digital;
6. B: **obtém a chave pública da Alice do seu chaveiro público** e verifica a sua confiança;
7. B: $v = V(m, pk, t)$ e aceita a mensagem caso $v =$'verifica'.

Note que é necessário que o Bob tenha previamente importado a chave da Alice para o seu chaveiro privado, sendo também necessário que esta seja já de confiança. Repare também que as mensagens e assinaturas **são codificadas e descodificadas em Base64** imediatamente **antes de serem enviadas e logo que são recebidas**, respetivamente.

1.7 Serviço de Confidencialidade
Confidentiality Service

Apesar de não ser um procedimento disruptivo, a forma de garantir a confidencialidade (cifra) de mensagens é interessante no PGP. Este procedimento está ilustrado na página seguinte, e incluem-se a seguir os passos que o determinam:

1. Alice (A): comprime a mensagem m para m';
2. A: $k \xleftarrow{r} \mathcal{K}$
   Alice gera uma chave de sessão simétrica aleatória;
3. A: $c_1 \leftarrow E_s(k, m')$;
   Alice cifra a mensagem comprimida com a chave de sessão;
4. A: $c_2 \leftarrow E_{pub}(pk_{Bob}, k)$
   Alice cifra a chave de sessão com a chave pública do Bob (que já deve estar no chaveiro);
5. A: $c'_1 = BASE64(c_1)$ e $c'_2 = BASE64(c_2)$
   Codifica c_1 e c_2 em base64;
6. A→ Bob (B): c'_1, c'_2
   Alice envia os dois criptogramas em Base64 ao Bob;
7. B: $c_1 = BASE64^{-1}(c'_1)$ e $c_2 = BASE64^{-1}(c'_2)$
   Bob descodifica os dois criptogramas;
8. B: $k \leftarrow D_{pub}(sk_{Bob}, c_2)$
   Bob obtém a chave de sessão decifrando c_2 com a sua chave privada;
9. B: $m' \leftarrow D_s(s, c_1)$
   Bob obtém a mensagem comprimida decifrando c_1 com a chave de sessão;
10. B: descomprime m' para obter m.



1.8 Ficha Técnica
Technical Sheet

Uma implementação compatível com a versão 7 do PGP (e.g., GnuPG versão 2.1.7) apresenta a seguinte ficha técnica, em termos de algoritmos usados nos diversos serviços:

Cifras de Chave Simétrica: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH, CAMELLIA128, CAMELLIA192, CAMELLIA256

Algoritmos de Compressão: Uncompressed, ZIP, ZLIB, BZIP2

Funções de Hash Criptográficas: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224

Cifras de Chave Pública: RSA, ElGamal, ELG

Esquemas de Assinatura Digital: RSA, DSA, ECDSA, EDDSA

2 Cartão do Cidadão Português

Portuguese Citizen Card

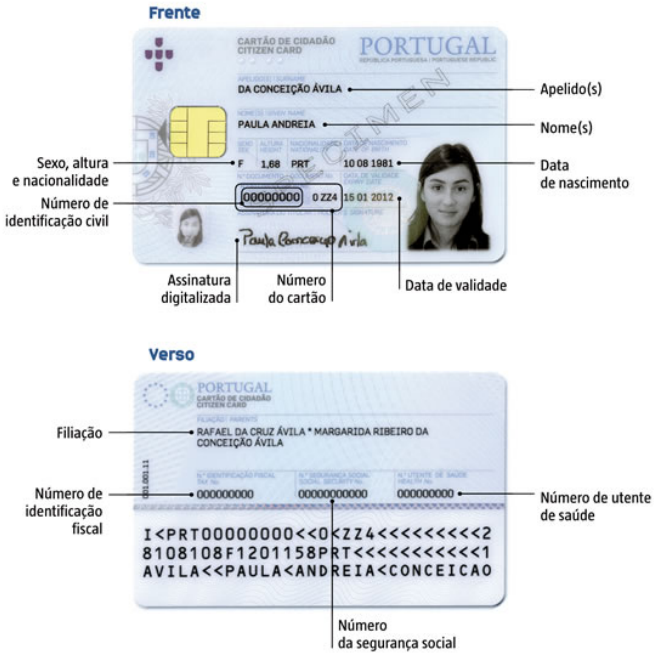
2.1 Introdução

Introduction

O **cartão do cidadão da república portuguesa** começou a ser emitido em **Fevereiro de 2007**. Como se discutirá adiante, veio **substituir 5 outros cartões** dos cidadãos portugueses, bem como **adicionar** algumas **funcionalidades** e fornecer **serviços novos**.

Segundo o próprio site de apresentação do cartão[2], este pode ser visto de várias formas e analisado de vários pontos de vista:

- Como **documento físico**, que permite ao cidadão identificar-se presencialmente de forma segura.
- Como **documento tecnológico**, que permite ao cidadão identificar-se perante serviços informatizados e autenticar documentos eletrónicos.
- Do **ponto de vista físico**, o Cartão de Cidadão tem um **formato *smartcard***.
- Do ponto de **vista visual**, o cartão **exibe**, na frente, a **fotografia e os elementos de identificação civil**. No **verso**, tem **os números de identificação dos diferentes organismos cujos cartões agrega e substitui, uma zona de leitura ótica e o *chip***.
- Do **ponto de vista eletrónico**, tem um ***chip* de contacto**, com **certificados digitais** (para autenticação e assinatura eletrónica), podendo ainda ter **a mesma informação do cartão físico, completada por outros dados**, designadamente a morada.



[2]A descrição nesta secção foi adaptada de `http://www.consuladoportugalparis.com/cartao-do-cidadao.html`.

Atualmente, o Cartão de Cidadão possui as seguintes funcionalidades:

- **Autenticação em serviços da Internet** – o cartão possibilita a autenticação em sites na Internet, utilizando para isso o certificado digital de autenticação presente no *chip* do cartão;
- **Assinatura eletrónica qualificada** - O cartão possibilita a assinatura eletrónica qualificada de documentos eletrónicos, sendo essa assinatura reconhecida em tribunal ou em todas a instâncias legais;
- **Armazenamento de notas pessoais do cidadão** – O cartão dispõe de um espaço no *chip* que permite o armazenamento livre de informação (texto) – espaço disponível limitado (pode ser, e.g., usado para guardar contactos de emergência);
- **Verificação de impressão digital** – funcionalidade que permite a verificação presencial do titular do documento. O Cartão de Cidadão **possui uma aplicação que compara a leitura recolhida no momento de uma impressão digital do cidadão com os dados da impressão digital existentes no cartão**. Ao realizar a operação de comparação no *chip*, garante-se a total confidencialidade das impressões digitais do cidadão, que em instante algum são transmitidas para fora do chip. Este serviço está disponível, em primeira instância, aos serviços de identificação da Administração Pública.

O cartão do cidadão da república portuguesa (início de emissão em Fevereiro de 2007) **substitui 5 outros cartões**:

1. Bilhete de Identidade;
2. Cartão do Contribuinte;
3. Cartão de Beneficiário da Segurança Social;
4. Cartão de Eleitor;
5. Cartão de Utente do Serviço Nacional de Saúde.

Note-se que **a multiplicidade dos números que identificam o cidadão** em cada um dos serviços **foi mantida**, estando estes números todos escritos (fisicamente) no cartão ou implicitamente embutidos neste.

2.2 O *Smartcard*

The Smartcard

O facto do cartão de cidadão ser um *smartcard* significa que contém um microprocessador eletrónico embutido, bem como memória, que são ativados quando colocados num leitor de cartões, e alimentados via conectores com energia elétrica. Este processador e memória servem para vários fins:

- **Guardar informação privada**, i.e. Informação que o utente pode **usar mas não divulgar** ou até conhecer, nomeadamente:
  - Uma **chave simétrica** de autenticação do titular;
  - Uma **chave privada**, de um **par de chaves assimétricas RSA**, que **serve para autenticar** o titular;
  - Uma **chave privada**, de um **par de chaves assimétricas RSA**, que serve **para produzir assinaturas digitais**.
- Guardar **informação pessoal para validação informática interna** da identidade do titular (e.g., o *template* da impressão digital é usado para validar uma impressão digital comunicada ao cartão (é o cartão que faz a validação **internamente**)).
- **Guardar informação reservada**, i.e. aquela que o titular conhece mas que apenas disponibiliza de forma fidedigna, via *smartcard*, a quem desejar ou tiver autorização para a obter independentemente da vontade do titular. Neste caso, estamos a falar da morada. O facto da morada estar guardada no *smartcard* permite, por exemplo, que o titular mude de morada e atualize os dados via aplicações informáticas, em vez de ter de produzir novo documento.
- Guardar **informação pública de grande dimensão, de difícil memorização**: fotografia do titular (em alta definição) e **certificados X.509v3 de chaves públicas do titular**, que podem ser usados para autenticar o titular ou a sua assinatura digital.
- Guardar **toda a informação do titular observável nas faces do cartão, mas em forma digital**.
- **Efetuar operações criptográficas** usando as chaves que fazem parte da sua informação privada.

2.3 *Personal Identification Numbers*

Personal Identification Numbers *(PINs)*

Operações realizadas pelo *smartcard* em nome do titular **necessitam que este indique um código secreto pessoal** (*Personal Identification Number* (PIN)). Em termos de **autorização de operações, há 3 PINs** diferentes para o cartão de cidadão:

- Um para autorizar a indicação da morada;
- Outro para **autenticação do titular**;
- Outro para **produzir assinaturas digitais**.

Estes PINs tornam o cartão do cidadão **mais pessoal** e adicionam o fator conhecido como *Something you know* na autenticação. Se o cartão for perdido ou extraviado propositadamente, quem o tem na sua posse tem **3 tentativas** para acertar no PIN antes de bloquear o *smartcard*. Adicionalmente, existe **um código de cancelamento de 8 dígitos** (fornecido com o próprio cartão do cidadão) que permite cancelar o dito cujo junto das autoridades competentes. Este código tem 8 dígitos para evitar ataques do tipo *Denial of Service* para cancelamentos, já que **se fosse pequeno, qualquer um podia tentar cancelar o cartão de outra pessoa**.

2.4 Autenticação de Dois Fatores usando o Cartão do Cidadão

Two-Factor Authentication using the Citizen Card

A autenticação usando o *smartcard* pode ser feita de duas formas:

1. Usando o *Europay, MarterCard and Visa Chip Authentication Program* (EMVCAP):
   Neste caso, insere-se o cartão num leitor pessoal, seleciona-se a opção para **criar uma *One-Time Password*** (OTP), **digita-se o PIN de autenticação** no terminal e **observa-se o código que este devolve**. O código pode ser **transmitido via telefone ou fax** para autenticar o dono do cartão. A entidade que está do outro lado **verifica se a OTP enviada é igual aquela que também gerou partindo dos mesmos valores.** Este método usa algoritmos de cifra simétrica (AES), uma chave simétrica (semente), e um contador para gerar as OTP e necessita que o servidor que autentica esteja sincronizado com o contador do utilizador.
2. Usando **um par de chaves assimétricas**, destinadas, apenas e só, para este efeito:
   Neste caso, **o PIN do titular tem de ser enviado para o *smartcard*** **cada vez** que for necessário usar a chave privada do par de chaves assimétricas para autenticação do titular. O *smartcard* tem e disponibiliza ainda o certificado X.509 com a chave pública de autenticação do titular, para que possa ser transmitido a quem queira verificar a correção e validade da chave privada de autenticação do titular.

A autenticação usando chaves RSA pode ser explicada da seguinte forma. Imagine que a Alice tem um cartão de cidadão e quer autenticar-se ao Bob:

1. Alice (A) $\rightarrow$ Bob (B): "Sou a Alice e quero autenticar-me!"
2. B: $n \xleftarrow{r} \mathcal{R}$
   Bob gera um número aleatório chamado desafio;
3. B $\rightarrow$A: r
   Bob envia o número à Alice;
4. A: $t \leftarrow S(pk, r)$
   A Alice assina o número que o Bob enviou (para isso usa o cartão de cidadão, e introduz o PIN de autenticação);

5. A $\rightarrow$B: t, Cert$_A$
   Alice envia ao Bob a assinatura daquele número e o seu certificado:

6. B: $v = V(t, r, pk_{Alice})$ e verifica o certificado (inclusive o caminho de certificação) e aceita a identidade da Alice caso $v =$'verifica' e o caminho de certificação validar.

Repare-se que **esta autenticação tem dois fatores**, porque **só a pessoa com o cartão da Alice (*something you have*) e com conhecimento do PIN de autenticação (*something you know*) é que conseguiria fazer a assinatura do número aleatório**. Este número muda para cada autenticação, pelo que não pode ser reutilizado por um atacante que esteja a escutar a comunicação.

2.5 Assinatura Digital
Digital Signature

O cartão do cidadão **possui um par de chaves assimétricas RSA para assinatura digital**. Possui igualmente o certificado X.509v3 relativo à chave pública usada para assinaturas, para que possa ser **disponibilizado à entidade que quer validar a assinatura digital**. **O PIN** da assinatura digital do titular **tem de ser enviado para o *smartcard* cada vez que for necessário usar a chave privada** dedica à cifragem.

Note que, por omissão, a funcionalidade de validação da assinatura digital **não vem cativada por defeito** para o cartão do cidadão: é possível produzir assinaturas desde o primeiro dia, mas as mesmas não podem ser validadas enquanto não se ativar a funcionalidade. A ativação desta funcionalidade tem de ser requerida presencialmente numa instituição autorizada para esse efeito. A respetiva chave é então retirada de uma **Lista de Revogação de Certificados**.

2.6 Cadeia de Certificação
Certification Path

O **certificado raíz** dos certificados do cartão do cidadão pertence a **uma empresa não portuguesa**: *GTE Corporation*. A maior parte do *software* que usa certificados X.509 utiliza esta esta raiz (é portanto provável que já esteja instalada no computador de qualquer utilizador).

Não obstante, qualquer cartão do cidadão **inclui também todos os certificados intermédios** (4 no total) que permitem validar o caminho de certificação. Isto significa que **qualquer pessoa** que possua um cartão do cidadão **pode validar as assinaturas de outra pessoa** que possui um cartão do cidadão (e só neste caso).

2.7 Usabilidade e Segurança
Security and Usability

Este cartão concretiza uma tentativa para **tornar a criptografia de chave pública utilizável a qualquer cidadão**. É claro que o **projeto de segurança** teve de ter em conta esse facto, mas também a possibilidade de o cartão ser roubado e usado por alguém mal intencionado. O **tamanho e quantidade dos PINs são o resultado dessa dicotomia**, bem como o **número de tentativas** que alguém pode submeter antes de o bloquear, **ajustado para 3** (a cada tentativa bem sucedida, o contador de erros é anulado.). Repare-se que **as chaves privadas e o *template* em alta resolução da impressão digital nunca chegam a sair do *smartcard***, por uma questão de segurança. Para desbloquear um PIN, é preciso deslocar-se à entidade autorizada, e indicar o código de desbloqueio, também fornecido com o cartão.

Ao todo, há **9 PINs diferentes**:

- Activação do *smartcard* (8) e da assinatura digital (4);
- Cancelamento do *smartcard* (8);
- PIN morada (4); autenticação (4); de assinatura (4);
- Desbloqueio do PIN de morada (4); de autenticação (4); de assinatura (4).

A interacção com o cartão do cidadão faz-se **através de *middleware***, nomeadamente de **bibliotecas normalizadas PKCS#11, publicamente disponíveis**. Para o caso do CC, a biblioteca `pteidpkcs11` permite o acesso ao cartão para operações criptográfica, enquanto que o `pteidlib` permite interagir com o cartão para realizar operações não criptográficas.

Nota: o conteúdo exposto na aula e aqui contido não é (nem deve ser considerado) suficiente para total entendimento do conteúdo programático desta unidade curricular e deve ser complementado com algum empenho e investigação pessoal.