**Instituto Superior de Engenharia de Lisboa**

**A logo with text on it

Description automatically generated**

**Segurança Informática**

***Trabalho 1***

***João Bonacho Nº A49437***

***André Gonçalves Nº A49464***

***Ana Carolina Pereira Nº A49470***

LEIC51D Grupo 07

Semestre de Inverno 2023/2024

28 de Outubro de 2023

**Parte 1**

**Exercício 1**

O esquema não cumpre os objetivos uma vez que parte do esquema MAC (*Message Authentication Code*) é utilizado como chave do esquema simétrico, acarretando possíveis vulnerabilidades de segurança. Um atacante pode utilizar os primeiros *bits* do MAC como chave e, de seguida, decifrar o criptograma enviado, dado que o MAC é enviado em claro. As chaves devem ser aleatórias e usadas por pouco tempo.

**Exercício 2**

A cifra simétrica é menos segura, mas mais rápida que a cifra assimétrica, necessitando apenas de, por exemplo, operações como XOR’s e deslocação de bits. Enquanto a cifra assimétrica é mais segura, mas mais lenta que a cifra simétrica, necessitando de, por exemplo, factorização de números primos e multiplicações.

Como habitualmente, os dados são textos em claro de maior dimensão, deve ser utilizada inicialmente uma cifra simétrica de forma a cifrar esses dados. Posteriormente deve ser utilizada cifra assimétrica para cifrar a chave simétrica utilizada, habitualmente de dimensão mais pequena em relação aos dados enviados.

**Exercício 3**

Os esquemas MAC (*Message Authentication Code*) e Assinatura Digital apresentam semelhanças e diferenças.

Relativamente às **semelhanças** ambas garantem integridade/autenticidade (a mensagem enviada é autêntica) e não garantem confidencialidade (a mensagem não é cifrada).

Relativamente às **diferenças**, o esquema MAC (*Message Authentication Code*) usa chaves simétricas, em oposição à Assinatura Digital que utiliza chaves assimétricas (pública e privada). Outra diferença consiste na validação, i.e., idealmente no MAC apenas o emissor pode autenticar e o recetor validar, enquanto na Assinatura Digital só o emissor pode autenticar, mas qualquer intermediário pode validar a mesma mensagem.

**Exercício 4.1**

Os certificados podem ser de confiança para um sistema (e.g.: *Sa*) e não de confiança para outro (e.g: *Sb*) dado que não depende do certificado, mas sim do sistema. Estes podem ter diferentes raízes de confiança que estão instaladas implicitamente.

**Exercício 4.2**

O mecanismo previsto nos certificados X.509 e no perfil PKIX consiste numa restrição ao uso do certificado (*Basic Constraints*) em que não pode haver certificados folha a meio da cadeia. Cada certificado tem uma extensão que indica se pode assinar certificados.

*“The cA boolean indicates whether the certified public key may be used to verify certificate signatures. If the cA boolean is not asserted, then the keyCertSign bit in the key usage extension MUST NOT be asserted. If the basic constraints extension is not present in a version 3 certificate, or the extension is present but the cA boolean is not asserted, then the certified public key MUST NOT be used to verify certificate signatures.”* **RFC 5280**

**Parte 2**

A screenshot of a computer

Description automatically generated**Exercício 5.1**

Figura - comandos Windows PowerShell para separar os bytes do cabeçalho e do corpo da imagem trab1.bmp

**A computer screen with a black background

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedExercício 5.2**

Figura 5 - comando Openssl para cifrar o corpo do BMP com algoritmo AES e modo de operação CBC

Figura 4 - comando Openssl para cifrar o corpo do BMP com algoritmo DES e modo de operação CBC

Figura 3 - comando Openssl para cifrar o corpo do BMP com algoritmo AES e modo de operação ECB

Figura 2 - comando Openssl para cifrar o corpo do BMP com algoritmo DES e modo de operação ECB

**Exercício 5.3**

**A screenshot of a computer

Description automatically generated**

Figura 6 - comandos Windows PowerShell para juntar os resultados anteriores com o cabelhaço

**A blue and purple sign with a lock and a padlock

Description automatically generatedA blue and grey sign with a lock

Description automatically generated**

Figura 8 - Imagem AES+ECB

Figura 7 – Imagem DES+ECB

**A blue background with small dots

Description automatically generatedA blue and purple speckled background

Description automatically generated**

Figura 10 – Imagem AES+CBC

Figura 9 – Imagem DES+CBC

**Observações e conclusões:**

1. Comparação: **DES+ECB** (fig. 7) e **DES+CBC** (fig. 9) ou **AES+ECB** (fig. 8) e **AES+CBC** (fig. 10)

Usando **primitivas iguais** e **modos de operação diferentes** são notórias as vantagens na escolha do modo de operação. Por exemplo, com o modo de operação ECB o atacante consegue interpretar a mensagem.

1. Comparação: **DES+ECB** (fig. 7) e **AES+ECB** (fig. 8) ou **DES+CBC** (fig. 9) e **AES+CBC** (fig. 10)

Usando **primitivas diferentes** e **modos de operação iguais** não são notórias as vantagens na escolha da primitiva.

**Exercício 6.3**

Numa cadeia de 100 blocos, se existir uma mudança de valor da transação no bloco 10, o *hash* do bloco 11 não vai coincidir com o *hash* calculado do bloco 10. Desta forma, a alteração seria detetada no processo de validação/comparação.

**Exercício 6.4**

Para poder fazer uma alteração legítima ao valor da transação, é necessário alterar o bloco 11, porque o *hash* do bloco 10 alterou. No entanto, como as outras seguintes dependem dos anteriores, todos os *hashes* dos blocos seguintes ao bloco 10 têm de ser recalculados para que a cadeia permaneça válida.