Departamento de Engenharia Eletrónica e Telecomunicações e de Computadores

Licenciatura em Engenharia Informática e de Computadores

Semestre de Inverno 2022/2023

Segurança Informática

1º Trabalho

Grupo 5

N. 47283 Ricardo Bernardino

N. 47249 Miguel Almeida

N. 45935 David Costa

1.

Considere um novo modo de operação definido por:

• Seja x = x1, . . . , xL a divisão nos blocos xi do texto em claro x. • RV é um vector aleatório, com a dimensão do bloco, gerado por cada texto em claro x.

• Seja yi = E(k)(xi ⊕ RV), para i = 1, . . . , L, onde E é a operação de cifra, k é a chave da cifra, ⊕ denota o ou-exclusivo bit a bit.

1.1. Defina o algoritmo de decifra para este modo de operação

DES em modo ECB *adicionar imagem do modelo*

Algoritmo de decifra -> xi = E(k)(yi) xor RV

1.2 Compare este modo de operação com o modo CBC quanto a: a) possibilidade de padrões no texto em claro serem evidentes no texto cifrado, b) capacidade de paralelizar a cifra.

a) O modo CBC é melhor que o modo ECB, visto que a possibilidade de padrões no texto em claro serem evidentes no texto cifrado é menor

por apenas ocorrer quando duas mensagens são iguais, com a mesma chave e vetor de iniciação. Enquanto que no modo EBC bastam dois blocos em claro iguais, cifrado com a mesma chave para obter texto cifrado igual.

b) O modo ECB é melhor por não existir paralelização no modo CBC. No modo CBC existe dependência do texto cifrado de um bloco para o outro (chaining

2.

O RFC 4880, "OpenPGP Message Format", especifica a cifra de mensagens (denominados objectos) como uma combinação entre esquemas assimétricos e simétricos: «[...] first the object is encrypted using a symmetric encryption algorithm. Each symmetric key is used only once, for a single object. A new "session key" is generated as a random number for each object (sometimes referred to as a session). Since it is used only once, the session key is bound to the message and transmitted with it. To protect the key, it is encrypted with the receiver's public key. [...]» Justifique a utilização desta abordagem com dois tipos de chave e explique sucintamente o processo de decifra de uma mensagem (object).

É usado esquema simétrico para cifrar a informação contida na mensagem e o esquema assimétrico para cifrar a chave usada na cifra da mensagem para posteriormente ser enviada por um meio de comunicação. Desta forma o custo computacional é significativamente maior. O processo de decifra é composto pela decifra da chave recorrendo à chave privada da pessoa que recebeu, com isso é obtida a chave simétrica usada na mensagem e feita a decifra da mesma mensagem.

3.

A engine classe Signature da JCA contém, entre outros, os seguintes métodos: void initSign(PrivateKey privateKey) void initVerify(PublicKey publicKey) void update(byte[] data) byte[] sign() boolean verify(byte[] signature)

<u>3.1.</u> Explique sucintamente o processamento realizado internamente no método sign com o objetivo de fazer a assinatura. Pode usar na explicação os métodos referidos que entenda relevantes.

Para se iniciar o procedimento de assinatura é primeiramente criada uma instância da classe Signature (com nome "signature" por exemplo),
usando o método "getInstance()" e passando como parâmetro o algoritmo de assinatura escolhido.
Após esta é realizado um "signature.initSign()" com a respetiva chave privada, para se realizar a inicialização do objeto. (Tornando,
segundo o modelo do JCA para objetos Signature, a instância inicializada.)
Finalmente é passado o byte array de mensagem à instância criada e inicializada previamente, usando o método "signature.update()", e finalizando o processo
de assinatura é usado o método "sinature.sign()" para retornar o array de bytes resultantes da operação de assinatura à mensagem.
O método "sign()" não só retorna um byte array como resultado, como também dá "reset" ao objeto instanciado da classe Signature de volta ao estado
em que se encontrava quando foi previamente inicializado para assinatura, ou seja quando se deu a chamada ao método "initSign()", isto dá-se como propósito de
realizar novas assinaturas com a mesma chave privada a mensagens futuras, caso o usuário o pretenda.

3.2. Considere que é instanciado um objeto Signature com a transformação "RSAwithMD5". Se em virtude de uma vulnerabilidade detectada na função de hash MD5 for computacionalmente fazível, dado x, obter x ′ ≠ x tal que MD5(x ′) = MD5(x), quais as implicações deste ataque para as assinaturas geradas/verificadas pelas transformação referida?

Este ataque é definido pelo ataque de pré-imagem, onde o atacante tenta encontrar uma segunda entrada que produza o mesmo hash de uma entrada específica.
As assinaturas geradas através do objeto instanciado com este hash estariam agora em perigo de segurança pois este atacante poderia replicar uma mensagem do assinante (ou seja com a sua assinatura) que seria lida pelo destinatário, e que tudo indicava que foi assinada pelo mesmo objeto.

4.

Considere os certificados digitais X.509 e as infraestruturas de chave pública:

4.1. Em que situações é que a chave necessária para validar a assinatura de um certificado não está presente nesse certificado?

Em assinaturas digitais, em que a validação (verificação) de um certificado é feita por qualquer entidade com a chave pública, que não está
no certificado. O certificado é unicamente assinado com a chave privada.

4.2. Porque motivo a proteção de integridade dos certificados X.509 não usa esquemas MAC (Message Authentication Code)?

O objetivo do certificado X509 é fornecer uma autenticidade forte em que uma chave privada é guardada em segurança para garantir uma
politica de não-repúdio de uma assinatura pela parte de uma entidade verificadora. O esquema MAC usa uma autenticidade simétrica,
tendo ambas as assinaturas e verificações realizadas com a mesma chave, o que significa que qualquer entidade que consiga verificar
uma assinatura também consegue-a reproduzir, isto não obedece à politica de não-repúdio, tornando possivel a disputa pela autoria de uma assinatura.
Com uma assinatura digital, usada no certificado X509, é usado um esquema assimétrico, onde a assinatura é realizado com uma chave privada,
guardada em segurança e não acessível a entidades verificadoras, e verificada por estas usando uma chave pública, sendo que 2 chaves distintas
e independentes de si mesmas previnem a replicação da assinatura.

4.3. Qual a diferença entre ficheiros .cer e ficheiros .pfx?

Os ficheiros .cer apenas contêm a chave pública (normalmente usado como trocas entre parceiros de integração, como empresas parceiras, ou entidades
com o propósito de partilhar o acesso à verificação de certas assinaturas), é usada para verificar tokens ou pedidos de autenticação de clientes
como por exemplo no handshake inicial SSL de um servidor com um cliente HTTP.
Os ficheiros .pfx contém ambas as chaves pública e privada para um certo certificado, não é partilhado normalmente fora de uma organização, é usado
para a realização de assinaturas ou autenticação de sistemas dentro da mesma organização.