Segurança Informática Semestre de Inverno 22/23

Grupo 16

Respostas às questões do enunciado

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

$$xi = (D(k)\ yi) \oplus RV$$

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.

Resposta:

a) Há possibilidade de existência de padrões em textos cifrados relativamente ao texto em claro, visto que há a possibilidade de existirem blocos maiores que a primitiva e o vetor usados para encriptar/cifrar. Não ocorre no caso do CBC, pois há dependência dos blocos anteriores em relação a cifra.

b) Existe uma vez que não há evidências que haja dependência entre os blocos que serão cifrados; ao contrário do CBC, onde a cifra dos blocos são serializáveis.

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).

Resposta:

Por motivos de segurança, caso alguém consiga, por diferentes meios, aceder a chave simétrica ou a chave-privada do recetor, não conseguirá decifrar a mensagem por não ter acesso a outra chave em falta. São necessárias as duas chaves para a decifra da mensagem, sendo assim garantida uma maior segurança.

Para que o recetor consiga realizar a decifra da mensagem, primeiramente necessitará de receber a chave-simétrica do emissor através de um canal seguro e decifrá-la com a sua chave-privada; de seguida deverá decifrar a mensagem cifrada com essa chave resultante da decifra.

3.1. 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.

Resposta:

Internamente, a função *sign()* produz uma assinatura a partir dos bytes resultantes dos sucessivos *updates*; o algoritmo usado para a assinatura é o passado à função *getInstance*() do objeto *Signature* em causa.

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?

Resposta:

Um ataque deste tipo iria ser fatal para a assinatura, visto que, caso o atacante altere a mensagem e o *hash* dessa mensagem alterada for igual ao *hash* gerado a partir mensagem original, tal alteração não seria detetada no processo de verificação da assinatura digital, violando assim o sistema de segurança.

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?

Resposta:

Uma vez que para a verificação da assinatura é necessário a chave pública do emissor, isso implicaria sempre a chave pública da autoridade de certificação acima, ou seja, isso aconteceria, na teoria, em todos os caso com a exceção dos certificados auto assinados (raízes de confiança).

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

Resposta:

A utilização do esquema MAC destruiria o conceito da utilização dos certificados X.509 pois os certificados digital baseiam-se no facto de que apenas a autoridade de certificação tenha conhecimento da chave utilizada para a criação da assinatura.

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

Respostas:

Um ficheiro ***.pfx*** inclui as chaves pública e privada para o certificado associado, podendo pode ser usado para assinatura digital de mensagens ou *tokens* de autorização ou para autenticação.

Já um ficheiro ***.cer*** inclui apenas a chave pública, não havendo transporte de chaves privadas, sendo mais usado para a verificação de *tokens*/criptogramas.