|  |  |
| --- | --- |
| C:\Users\PM\Desktop\RC-Projecto-Rede.bmp | **Instituto Superior de Engenharia de Lisboa**  Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores  **Segurança Informática, 5º Semestre** |

Segurança Informática

*1ª Série*

Docente: Pedro Félix

|  |  |  |  |
| --- | --- | --- | --- |
| Elaborado por: |  |  |  |
|  |  |  |  |
| André Cunha  Pedro Marinho | G06  G006 | LI51N  LI51N | nº31612  nº36122 |

Lisboa, 05 de Maio de 2013

Índice

[1. Respostas ao enunciado 3](#_Toc401692250)

# Respostas ao enunciado

1. *Considere um esquema de cifra simétrica baseado numa primitiva de cifra em bloco com n = 128 (dimensão do bloco) e l = 256 (dimensão da chave), e que utiliza o modo de operação CBC com padding PKCS#5.  
   Qual a dimensão do criptograma resultante da cifra duma mensagem com 256 bits.*

R: A dimensão do criptograma resultante da cifra, nas condições do enunciado, é de 512 bits. Em que 128 bits correspondem ao Initialization Vector (IV), 256 bits de mensagem e 128 bits de PKCS.

1. *A obtenção simultânea de integridade e autenticidade pode ser conseguida através da utilização de esquemas MAC e de cifra simétrica, usando uma das seguintes técnicas: Encrypt-then-Authenticate ou  
   Authenticate-then-Encrypt. Qual destas técnicas é imune ao ataque de Vaudenay?*

R: A técnica *Encrypt-then –Authenticate* (EtA) é imune ao ataque de Vaudenay.

A mensagem é primeiro encriptada e depois assinada. Deste modo a validação é feita antes da desencriptação. Isso impede que se monte uma mensagem uma vez que a assinatura não vai corresponder. Assim o criptograma nem sequer e desencriptado, sendo descartado imediatamente.

1. *Quais as semelhanças e as diferenças entre um esquema assimétrico de assinatura digital e um esquema  
   MAC? Quais os critérios de decisão para seleccionar um deles?*

R: A única diferença é que a assinatura digital usa uma chave privada para assinar e uma chave pública para verificar, enquanto o MAC usa a mesma chave para assinar e verificar.

1. *No contexto das infra-estruturas de chave pública, apresente uma técnica para proteger as end-entities de  
   ataques às autoridades de certificação.*

R: Usando as CRL(certificate revocation list) e possível obter a lista de certificados inválidos, podendo assim recusar a autoridade de um certificado gerado com base num ataque.

1. *No contexto das infra-estruturas de chave pública baseadas em certificados X.509  
   5.1. Qual a relação entre a chave pública presente num certificado e a chave usada na assinatura deste?*

R: São chaves do mesmo par.

A chave pública da entidade assinante é enviada junto do certificado, isto permite que uma vez reconhecida a autoridade do certificado podemos validar documentos enviados pelo assinante (assinada com o chave privada), ou encriptar algo que apenas a chave privada do assinante pode desencriptar.

*5.2. Quais as consequências se uma aplicação consumidora de certificados ignorar a extensão basic constraints?*

R: A extensão basic constraint é utilizada para definir o tipo de certificado. Ignorar esse campo implicaria que podemos estar a reconhecer erradamente a autoridade de um certificado para realizar determinadas acções. Por exemplo, poderíamos utilizar um certificado destinado a ser end-entity como um certificado de autoridade intermédio.

*5.3. Em que circunstância um certificado X.509 contém uma chave privada?*

R: Um certificado X.509 nunca contém uma chave privada. A chave privada está apenas presente num ficheiro pfx à parte.

1. Seja *h k* : *{*0*,* 1*} ∗ → {*0*,* 1*} k* a função de *hash* definida por *h k*(*x*) = *y* 1 *. . . y k*  
   onde *y* 1 *. . . y* 160 = **SHA1**(*x*).  
   Sejam *m* 1 e *m* 2 os programas Java definidos nos ficheiros BadApp.java e GoodApp.java (presentes em  
   anexo ao enunciado). Dois programas *m* e *m 0* dizem-se equivalentes (*m ≡ m 0*) se a sua execução produz  
   o mesmo resultado observável.  
   6.1. Calcule *h k*(*m* 1) e *h k*(*m* 2) para *k* = 8*,* 16*,* 32.  
   6.2. Realize uma aplicação para encontrar um programa *m 0* tal que *h k*(*m 0*) = *h k*(*m* 2) e *m 0 ≡ m* 1.  
   Considere *k* = 8*,* 12*,* 16. Qual o número médio de operações *h k* necessário para encontrar a colisão?  
   6.3. Realize uma aplicação para encontrar um par (*m 0*1*, m 0*2) tal que *h k*(*m 0*1) = *h k*(*m 0*2)*, m 0*1 *≡ m*1 e  
   *m 0*  
   2 *≡ m*2. Considere *k* = 8*,* 16*,* 32. Qual o número médio de operações *h k* necessário para encontrar  
   a colisão?

R: Ver código em anexo.

1. Realize uma aplicação consola para assinar e verificar objectos *JSON Web Token* (JWT) [1], transportados  
   numa estrutura *JSON Web Signature* (JWS) [2]. A aplicação deve no mínimo suportar assinatura digital  
   com os algoritmos “RS256” e “HS256” [3].

R: Ver código em anexo.