Criptografia assimétrica

SIO

deti universidade de aveiro
departamento de eletrónica, telecomunicações e informática

João Paulo Barraca

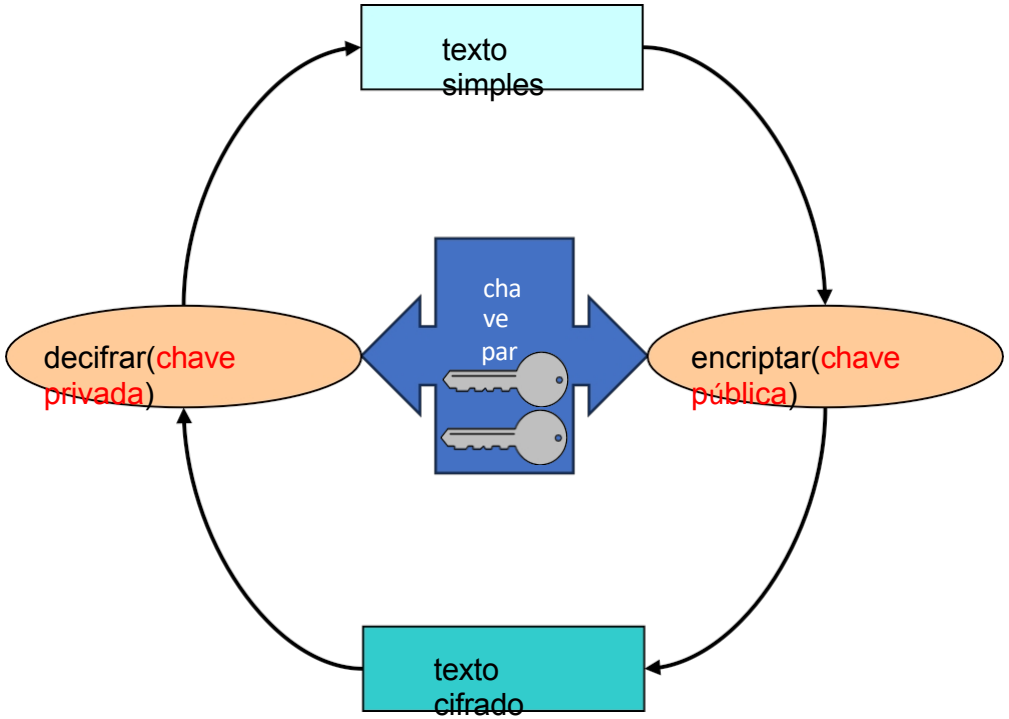
Cifras assimétricas (de bloco)

- Utilizar pares de chaves
  - **Uma chave privada**: pessoal, não transmissível
  - **Uma chave pública**: disponível para todos

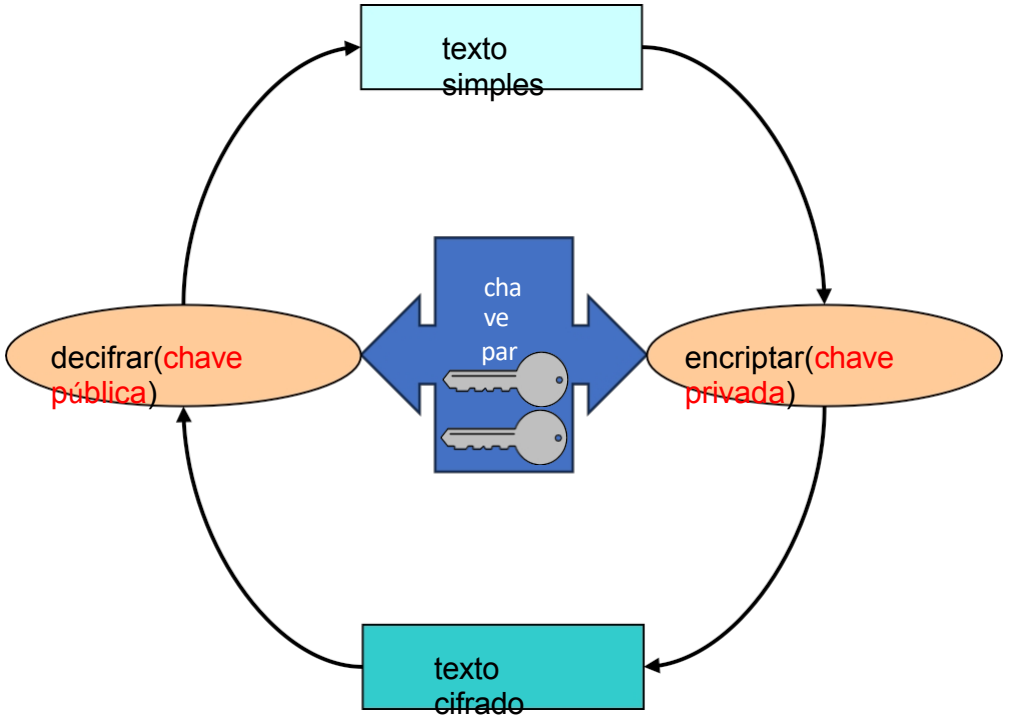
- Permitir
  - Confidencialidade sem qualquer troca prévia de segredos
  - Autenticação
    - De conteúdo (integridade dos dados)
    - Da origem dos dados (autenticação da fonte ou assinatura digital)

Operações de uma cifra assimétrica

Confidencialidade



Autenticidade

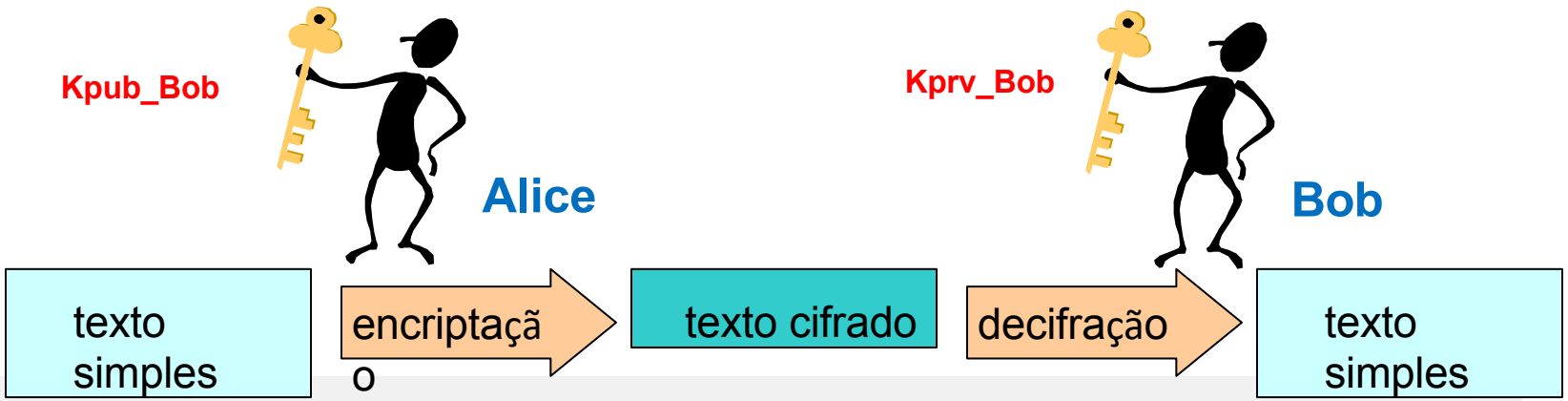


Casos de utilização: comunicação confidencial

- Comunicação segura com um alvo (Bob)
  - Alice cifra o texto simples **P** com a chave pública de Bob K_{pub_Bob}

    Alice: C = $\{P\}K_{pub\_Bob}$
  - O Bob desencripta o texto **cifrado C** com a sua chave privada K_{prv_Bob}

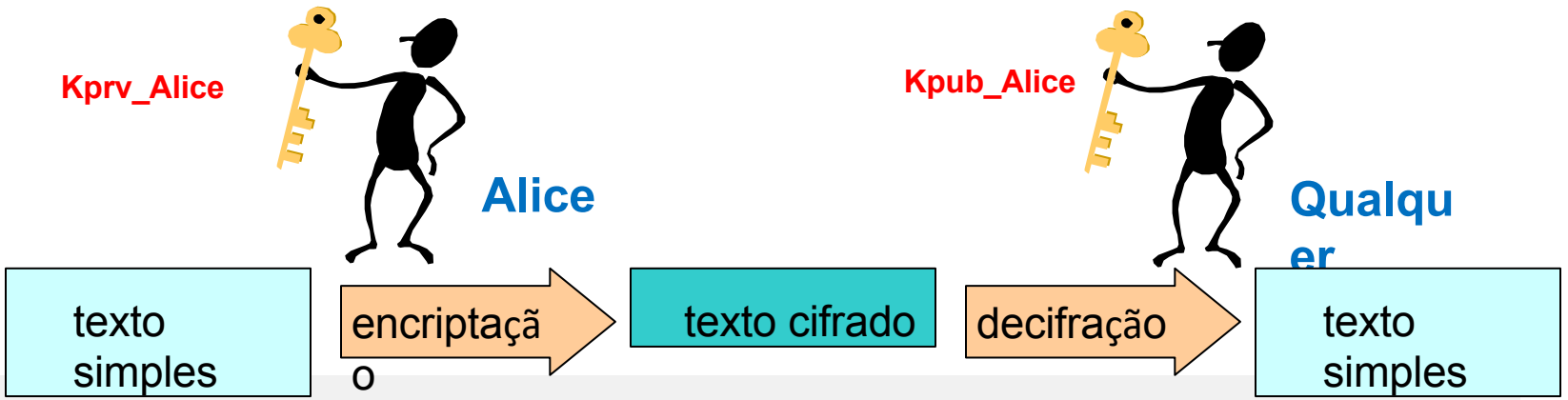
    Bob: P'= $\{C\}K_{prv\_Bob}$
  - **P'** deve ser igual a **P** (requer verificação através do controlo da integridade)
  - K_{pub_Bob} precisa de ser conhecido por Alice



Casos de utilização: comunicação autenticada

- Autenticar a comunicação de Alice
  - Alice cifra o texto simples **P** com a sua chave privada $_{Kprv\_Alice}$

    Alice: C = $_{\{P\}Kprv\_Alice}$
  - Qualquer pessoa pode decifrar o texto **cifrado C** com a chave pública de Alices $_{Kpub\_Alice}$ **Qualquer pessoa: P'=** $_{\{C\}Kpub\_Alice}$
  - Se **P'** = **P**, então **C** é a assinatura de Alice de **P**
  - $_{Kpub\_Alice}$ tem de ser conhecido pelos verificadores de mensagens



Cifras assimétricas

Questões

- Vantagens
  - Constituem um mecanismo de autenticação fundamental
  - Permitem explorar caraterísticas que não são possíveis com cifras assimétricas
- Desvantagens
  - Desempenho: 2 ou 3 ordens de grandeza em relação ao AES
  - Muito ineficaz e consome muita memória: Chaves grandes
- Problemas
  - Distribuição fiável de chaves públicas: como saber se a chave pública é a correta?
  - Tempo de vida dos pares de chaves: Como garantir que podemos lidar com chaves perdidas/depreciadas/apagadas?

Cifras assimétricas

Visão geral

- Abordagens: problemas matemáticos complexos
  - **Logaritmos discretos** de números grandes
  - **Factorização** de números **inteiros** grandes

- Algoritmos mais comuns
  - RSA
  - ElGamal
  - Curvas elípticas (ECC)

- Outras técnicas com pares de chaves assimétricas

  -Diffie-Hellman (acordo de chaves)

RSA

Rivest, Shamir, Adelman, 1978

- Chaves: Privado: (d, n) Público: (e, n)

- Encriptação de chave pública (confidencialidade) de **P**
  - **$C = P^e \bmod n$**
  - **$P = C^d \bmod n$**

- Encriptação de chave privada (autenticidade) de **P**
  - **$C = P^d \bmod n$**
  - **$P = C^e \bmod n$**

P, C são números!
A mensagem é convertida de/para números

$0 \leq P, C < n$

RSA

Rivest, Shamir, Adelman, 1978

- Complexidade computacional: **Logaritmo discreto** e **factorização de números inteiros**

- Seleção de chaves
  - Grande **n** (centenas ou milhares de bits)
  - **n = p × q**, sendo **p** e **q** números primos grandes (secretos)
  - Selecionar um **e** co-primo com **(p-1) × (q-1)**
  - Calcular **d** tal que **e × d ≡ 1 [mod (p-1) × (q-1)]**
  - Rejeitar **p** e **q**
  - O valor de **d** não pode ser calculado a partir de e e **n**
    - Apenas de **p** e **q**

coprimo → gcd(a, b) = 1

× → multiplicação

mod → operação de módulo

≡ → congruência modular

a ≡ b mod n iff rem(a,n) = rem(b,n)

Brincar com o RSA

- p = 5q = 11 (números primos)
  - n = p x q = 55
  - (p-1) x (q-1) = 40
- e = 3 (chave pública = e, n)
  - Coprimo de 40
- d = 27 (chave privada = d, n)

  - e x d ≡ 1 (mod 40) -> d x e mod 40 = 1 -> (27 x 3) mod 40 = 1
- Para uma mensagem a cifrar, P = 26 (note que P, C ∈ [0, n-1])
  - $C = P^e \bmod n$ $= 26^3 \bmod 55 = 31$
  - $P = C^d \bmod n$ $= 31^{27} \bmod 55 = 26$

Encriptação híbrida

- Combina criptografia simétrica com criptografia assimétrica
  - Utilizar o melhor dos dois mundos, evitando problemas
  - Cifra assimétrica: Utiliza chaves públicas (mas é lenta)
  - Cifra simétrica: Rápida (mas com métodos de troca de chaves fracos)

- Método:
  - Obter K_{pub} do recetor
  - Gerar um aleatório K_{sym}
  - Calcular **$C1 = E_{sym} (K_{sym} , P)$**
  - Calcular **$C2 = E_{asym} (K_{pub} , K)_{sym}$**
  - Enviar **C1 + C2**
    - C1 = Texto cifrado com chave simétrica
    - C2 = Chave simétrica cifrada com a chave pública do recetor
      - Pode também conter o IV

Randomização de encriptações assimétricas

- O RSA é um algoritmo determinístico: mensagens iguais resultam em resultados iguais

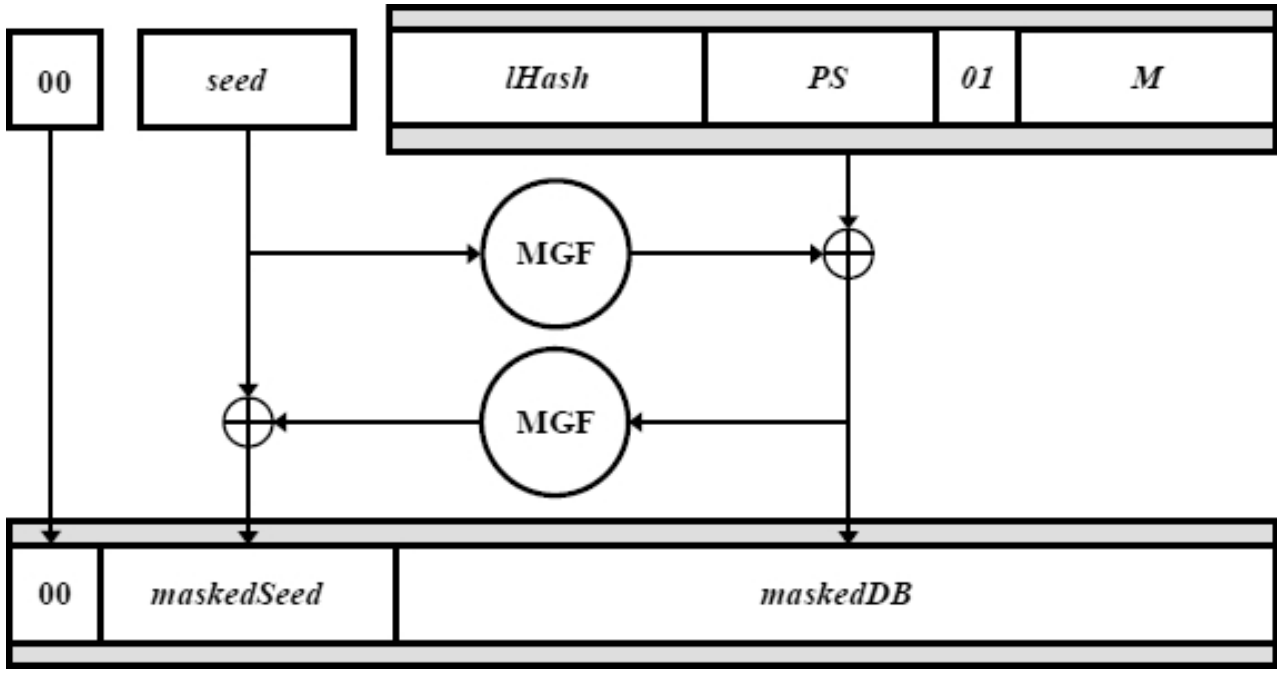
- O que precisamos: Resultado não determinístico de encriptações assimétricas
  - **N** encriptações do mesmo valor, com a mesma chave, devem produzir N resultados diferentes
  - **Objetivo: evitar a descoberta de valores encriptados por tentativa e erro**

- Abordagens
  - Concatenação do valor a encriptar com dois valores
  - Um fixo (para controlo da integridade)
  - Um aleatório (para aleatorização)

Randomização de encriptações assimétricas

OAEP (Optimal Asymmetric Encryption Padding)

- iHash: digest over Label
- semente: valor aleatório
- PS: zeros
- M: texto simples
- MGF: Função de geração de máscaras
  -Semelhante ao Hash, mas com tamanho variável



Acordo de chaves Diffie-Hellman (1976)

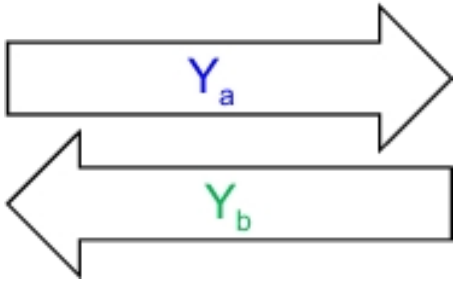
q (primo grande)

α (raiz primitiva mod q)

a = aleatório

$Y_a = \alpha^a \bmod q$

$Kab = Y_b^a \bmod q$



$Kab = K_{ba}$

b = aleatório

$Y_b = \alpha^b \bmod q$

$Kba = Y_a^b \bmod q$

Acordo de chaves Diffie-Hellman (1976)

a = aleatório

$Y_a = \alpha^a \bmod q$

$K_{ac} = Y_c^a \bmod q$

Y_a

$Y\ Y_c$

c = aleatório

$Y_c = \alpha^c \bmod q$

$K_{ca} = Y_a\ c \bmod q$

$K_{cb} = Y_b^c \bmod q$

Y_b

c

b = aleatório

$Y_b = \alpha^b \bmod q$

$K_{bc} = Y_c^b \bmod q$

Criptografia de curva elíptica (ECC)

- As curvas elípticas são funções específicas
  - Têm um gerador (G)
  - Uma chave privada K_{prv} é um número inteiro com um máximo de bits permitido pela curva
  - Uma chave pública K_{pub} é um ponto $(x,y) = K_{prv} \times G$
  - Dado K_{pub} , deve ser difícil adivinhar K_{prv}

- Curvas
  - Curvas NIST (15)
    - P-192, P-224, P-256, P-384, P-521
    - B-163, B-233, B-283, B-409, B-571
    - K-163, K-233, K-283, K-409, K-571

Outras curvas

- Curva25519 (256 bits)
- Curva448 (448 bits)

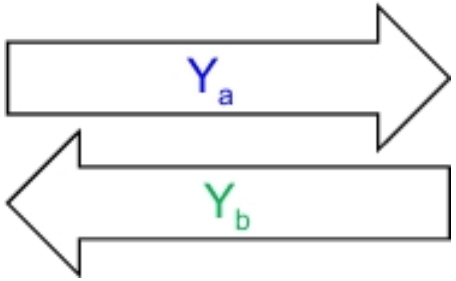
ECDH: DH com ECC

Curva ECC → G

a = aleatório

$Y_a = a\ G$

$K_{ab} = a\ Y_b$



b = aleatório

$Y_b = b\ G$

$K_{ba} = b\ Y_a$

$K_{ab} = K_{ba}$

Encriptação de chave pública ECC

Combina a encriptação híbrida com a ECDH

- Obter K_{pub_recv} do recetor
- Gerar um K_{prv_send} aleatório e o K_{pub_send} correspondente
- Calcular $K_{sym} = K_{prv\_send}\ K_{pub\_recv}$

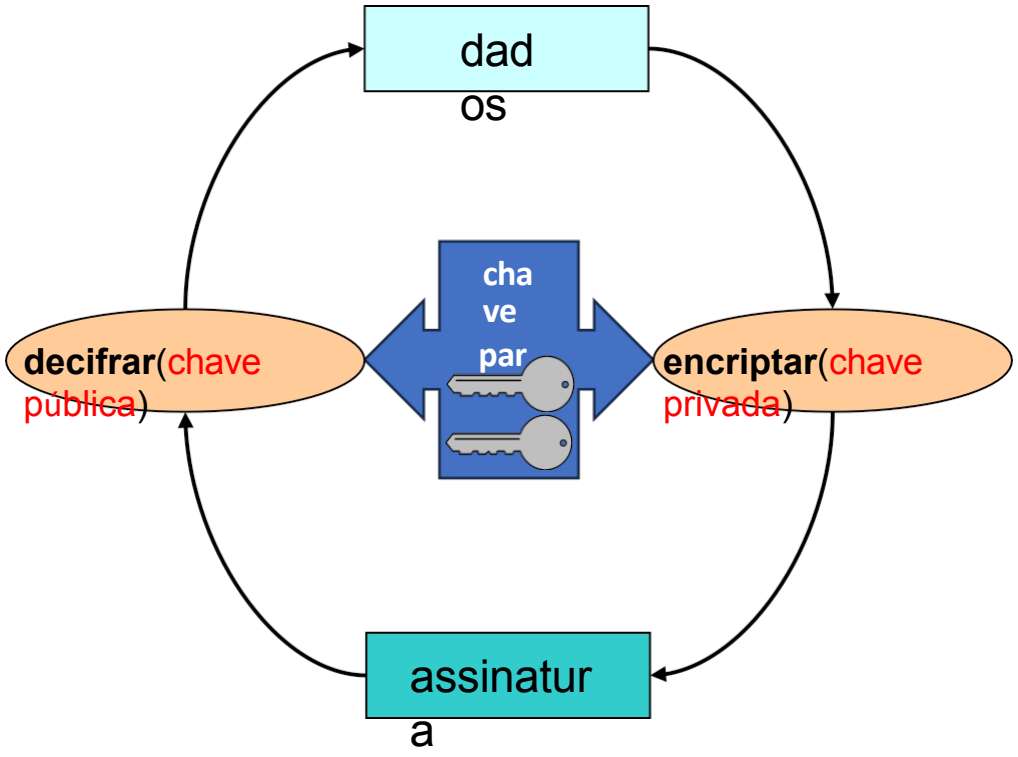
- $C = E(P, K)_{sym}$

- Enviar $C + K_{pub\_send}$
- O recetor calcula $K_{sym} = K_{pub\_send}\ K_{prv\_recv}$

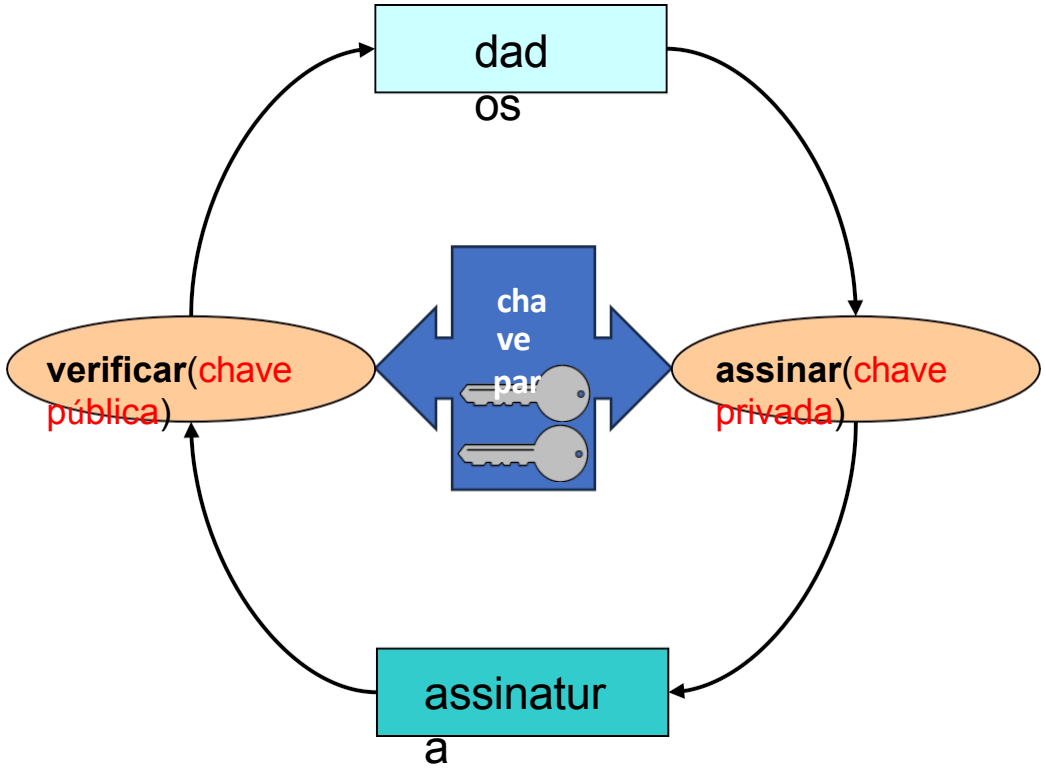
- $P = D(C, K)_{sym}$

Assinaturas digitais

Encriptar/Desencriptar (RSA)



Assinar/Verificar (ElGamal, EC)



Operações com chaves privadas

- Autenticar o conteúdo de um documento
  - Garantir a sua integridade (não foi alterado)
- Autenticar o seu autor
  - Garantir a identidade do criador/originador
- Impedir o repúdio da carga encriptada
  - Não repúdio
  - Os autores genuínos não podem negar a sua autoria
    - Apenas o autor identificado poderia ter gerado uma determinada carga útil
    - Porque só o autor tem a chave privada

Assinaturas digitais

- Autenticar o conteúdo de um documento
  - Garantir a sua integridade (não foi alterado)
- Autenticar o seu autor
  - Garantir a identidade do criador/originador
- Impedir o repúdio de assinaturas
  - Propriedade de não repúdio
  - Os autores genuínos não podem negar a sua autoria
    - Apenas o autor identificado poderia ter gerado uma determinada assinatura

Considerações práticas

- A encriptação com chave privada é vital para a autenticação
  - Apenas o autor o pode fazer, todos o podem verificar

- Mas... o envio de textos autenticados seguros exigirá dois (lentos) encriptações
  - Lembre-se: As cifras assimétricas são lentas e ineficientes

- Abordagem preferida: **Encriptar Hash(T), criando assinaturas digitais**

Assinaturas digitais

- Abordagens
  - Função de digestão do texto (apenas para o desempenho)
  - Encriptação/desencriptação assimétrica ou assinatura/verificação

Assinatura:

$A_x(doc) = informação + E(K_x^{-1}, digerir(doc + info))$

$A_x(doc) = informação + S(K_x^{-1}, digerir(doc + info))$

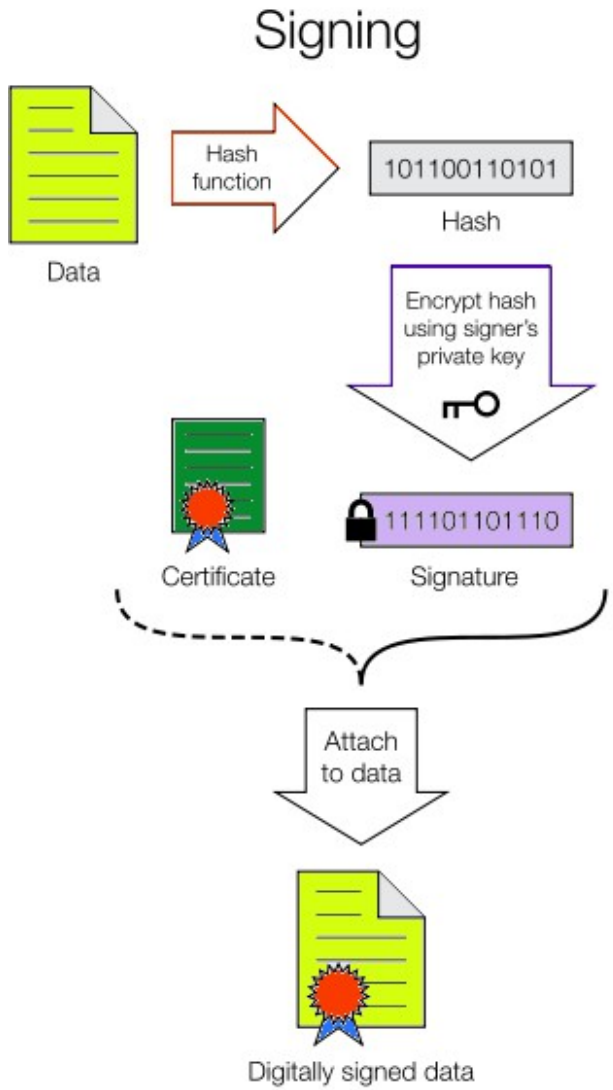
info = contexto de assinatura, identidade do signatário, K_x

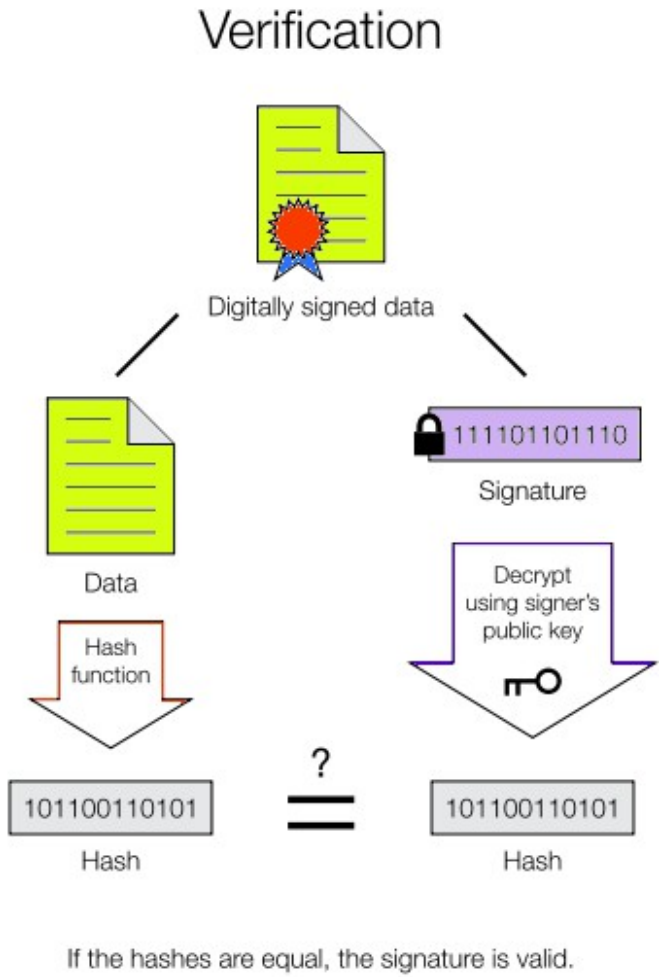
Verificação:

$D(K_x , A_x (doc)) \equiv digest(doc + info)$

$V(K_x , A_x (doc), doc, info)$ → Verdadeiro / Falso

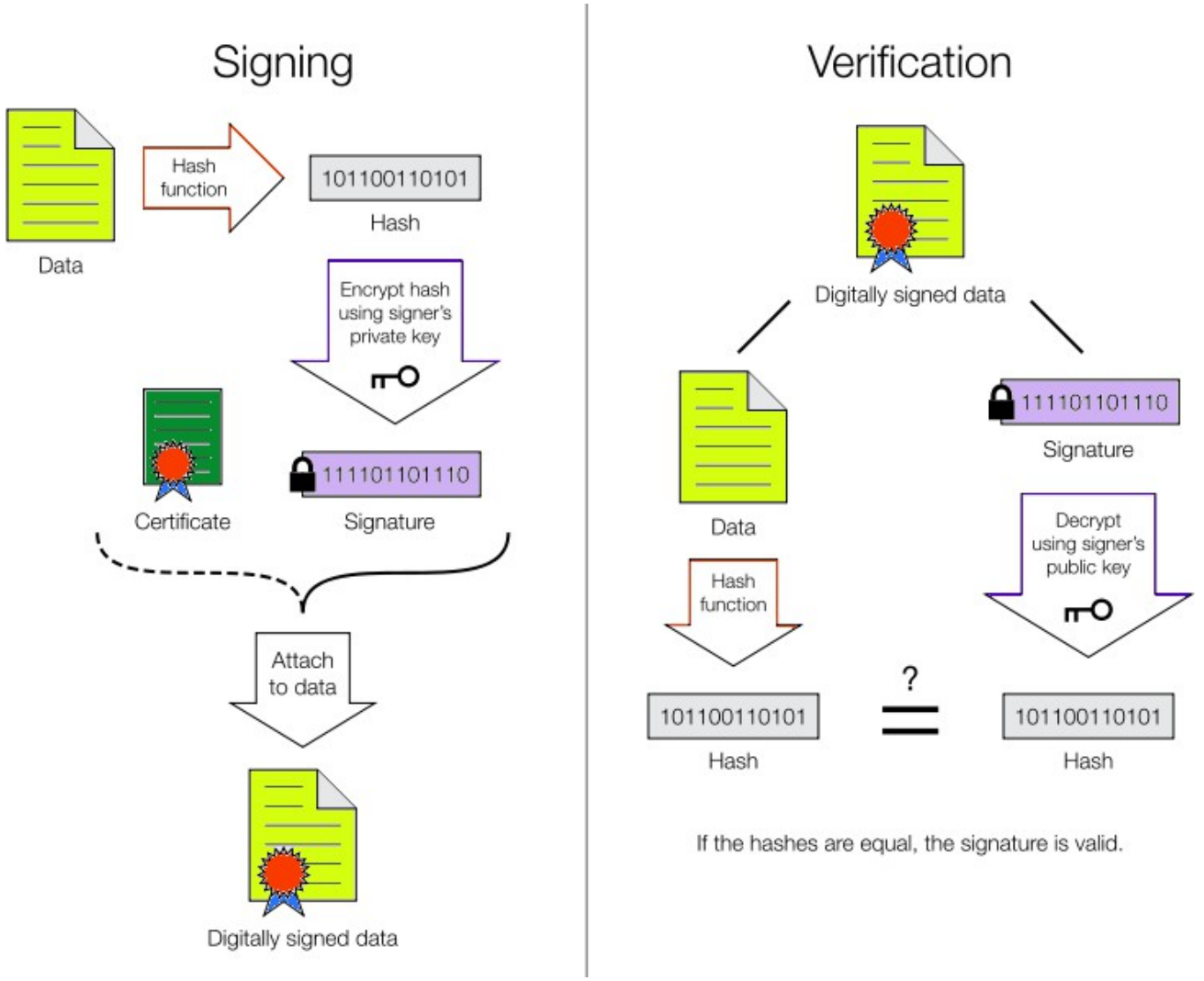
Assinaturas de encriptação / desencriptação





wikipedia, http://en.wikipedia.org/wiki/Digital_signature

Assinaturas de encriptação / desencriptação



Assinatura digital numa mensagem de correio eletrónico

Conteúdo de várias partes, assinatura com certificado

```
De - Sex Oct 02 15:37:14 2009
[...]
Data: Fri, 02 Oct 2009 15:35:55 +0100
De: Utilizador A <usera@domain.com>
MIME-Version: 1.0
Para: Utilizador B
<userb@domain.com> Assunto:
Teste
Content-Type: multipart/signed; protocol="application/x-pkcs7-signature"; micalg=sha1; boundary="------------------------------------ms050405070101010502050101"

Esta é uma mensagem assinada criptograficamente no formato MIME.

--------------------------ms050405070101010502050101
Content-Type: multipart/mixed;
  boundary="--------------------------060802050708070409030504"

Esta é uma mensagem com várias partes em formato MIME.
--------------------------060802050708070409030504
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: quoted-printable

Corpo do correio

--------------------------060802050708070409030504—
--------------------------ms050405070101010502050101
Content-Type: application/x-pkcs7-signature; name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7s"
Content-Description: Assinatura criptográfica S/MIME

MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAQAAoIIamTCCBUkwggSyoAMCAQICBAcnIaEwDQYJKoZIhvcNAQEFBQAwdTELMAkGA1UEBhMCVVMxGDAWBgNV
[...] KoZIhvcNAQEBBQAEgYCofks852BV77NVuww53vSxO1XtI2JhC1CDlu+tcTPoMD1wq5dc5v40Tgsaw0N8dqgVLk8aC/CdGMbRBu+J1LKrcVZa+khnjjtB66HhDRLrjmEGDNttrEjbqvpd2QO2
vxB3iPTlU+vCGXo47e6GyRydqTpbq0r49Zqmx+IJ6Z7iigAAAAAAAA==

--------------------------ms050405070101010502050101--
```

Assinaturas digitais em kernel.org