Laboratório de Redes

UFPB

Centro de Informática
UFPB

Cap. 2: Ferramentas criptográficas

Iguatemi E. Fonseca

iguatemi@ci.ufpb.br

Agenda

- Confidencialidade com cifração simétrica
- Autenticação de mensagens e funções de Hash
- Criptografia de chave pública ou criptografia assimétrica
- Aplicações de criptografia de chave pública:
  - assinaturas digitais
  - gerenciamento de chave

Confidencialidade com cifração simétrica

Cifração simétrica

- A técnica universal para prover confidencialidade para dados transmitidos ou armazenados é a criptografia simétrica
- **Sinônimos**: cifração convencional, cifração de chave única ou cifração de chave privada
- Componentes de uma técnica de criptografia simétrica:
  - **Texto as claras**: é a mensagem ou dados originais dado como entrada do algoritmo de cifração
  - **Algoritmo de cifração**: O algoritmo executa várias substituições e transformações no texto as claras
  - **Chave secreta**: também fornecida como entrada para o algoritmo de cifração
  - **Texto cifrado**: mensagem embaralhada produzida como saída
  - **Algoritmo de decifração**: é o algoritmo de cifração executado ao contrário

Cifração simétrica

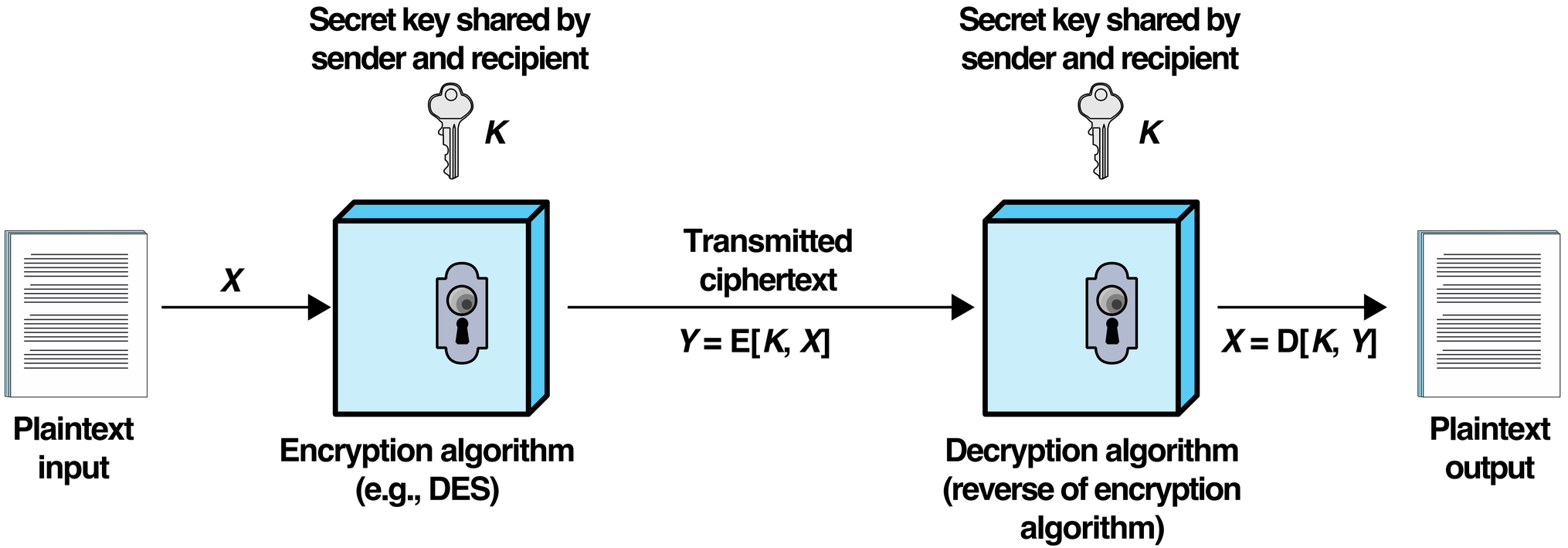


Figure 2.1 Simplified Model of Symmetric Encryption

Cifração simétrica

- Há dois requisitos para a utilização segura da cifração simétrica:
  - **Precisamos de um algoritmo de cifração forte**: deve ser incapaz de decifrar o texto cifrado ou descobrir a chave mesmo que se esteja de posse de vários textos cifrados juntamente com o texto às claras que produziu cada texto cifrado
  - **Remetente e destinatário devem obter cópias da chave secreta de maneira segura e mantê-las em segurança**: Se alguém conseguir descobrir a chave e conhecer o algoritmo, o efeito é catastrófico, i.e., toda comunicação que usar essa chave pode ser lida
- Ataques genéricos:
  - criptoanálise
  - ataque de força bruta

Cifração simétrica

Tabela 2.1 Tempo médio requerido para busca exaustiva de chave

Tamanho da chave (bits)	Número de chaves possíveis	Tempo requerido em 1 decifração/μs	Tempo requerido em 10^6 decifrações/μs
32	$2^{32} = 4{,}3 \times 10^9$	2^{31} μs × 35,8 minutos	2,15 milissegundos
56	$2^{56} = 7{,}2 \times 10^{16}$	2^{55} μs = 1.142 anos	10,01 horas
128	$2^{128} = 3{,}4 \times 10^{38}$	2^{127} ms = $5{,}4 \times 10^{24}$ anos	$5{,}4 \times 10^{18}$ anos
168	$2^{168} = 3{,}7 \times 10^{50}$	2^{167} μs = $5{,}9 \times 10^{36}$ anos	$5{,}9 \times 10^{30}$ anos
26 caracteres (permutação)	$26! = 4 \times 10^{26}$	2×10^{26} μs = $6{,}4 \times 10^{12}$ anos	$6{,}4 \times 10^{6}$ anos

Algoritmos Simétricos de Cifração de Bloco

- Os algoritmos de cifração simétricos mais comumente usados são cifras de bloco
- Uma cifra de bloco processa o texto às claras fornecido como entrada em blocos de tamanho fixo e produz um bloco de texto cifrado de tamanho igual para cada bloco de texto às claras
- Algoritmos simétricos mais importantes:
  - Data Encryption Standard (DES)
  - Triple DES (DES triplo)
  - Advanced Encryption Standard (AES)

Data Encryption Standard (DES)

- Adotado em 1977 pelo NIST
- É o algoritmo de criptografia mais utilizado em sistemas legados
- Toma como entrada um **bloco de texto às claras de 64 bits** e uma **chave de 56 bits**, para produzir um **bloco de texto cifrado de 64 bits**
  - **Quantas chaves diferentes o DES pode ter?**
- Duas preocupações: i) fraqueza do algoritmo ii) tamanho da chave
- Em 1998, *Electronic Frontier Foundation* (EFF) conseguiu quebrar o algoritmo DES com uma máquina que custou 250 mil dólares para ser construída

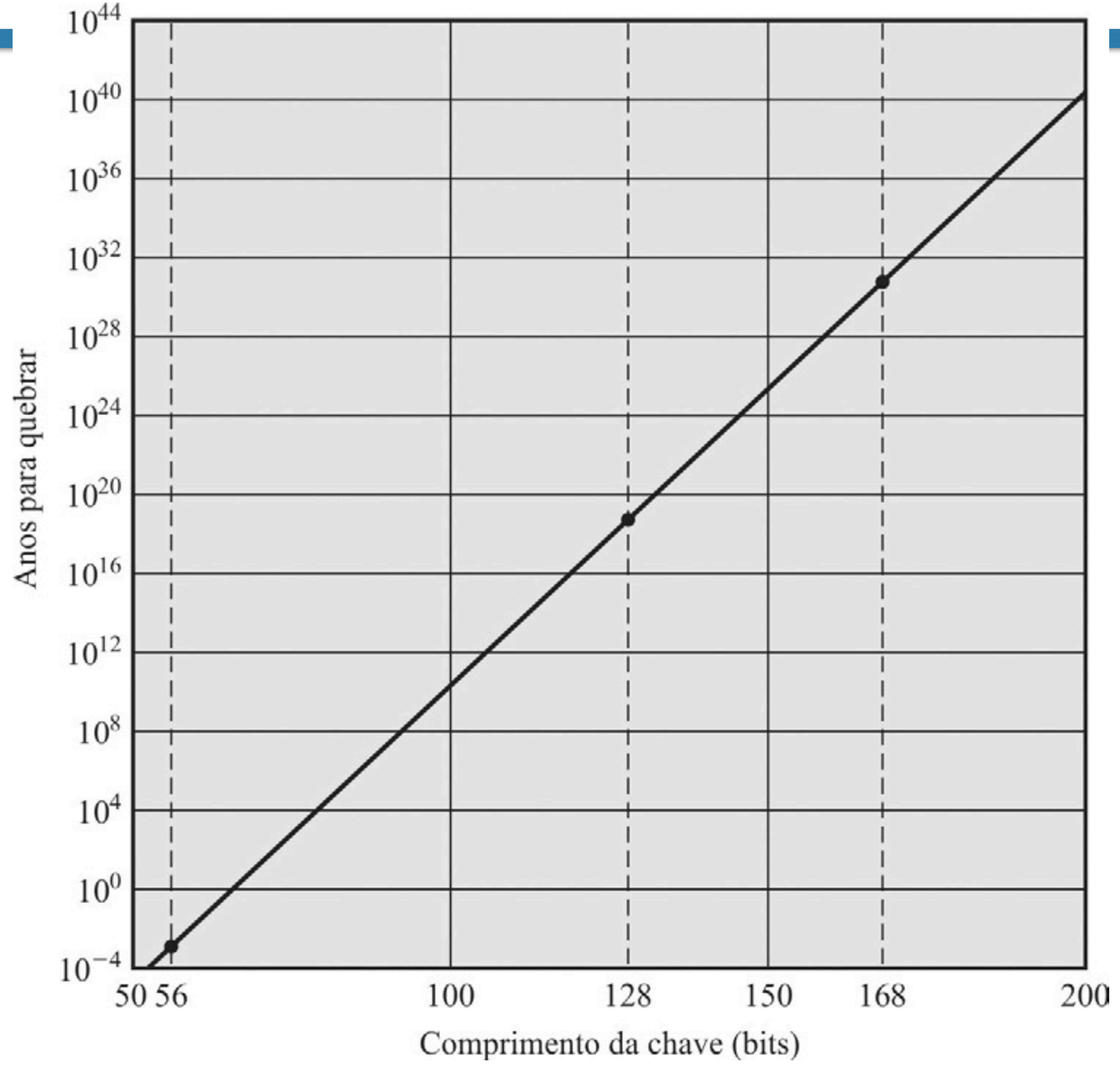


FIGURA 2.2 Tempo para quebrar um código (considerando 10^6 decifrações/μs) O gráfico considera que um algoritmo de cifração simétrico é atacado usando uma abordagem de força bruta de testar todas as chaves possíveis.

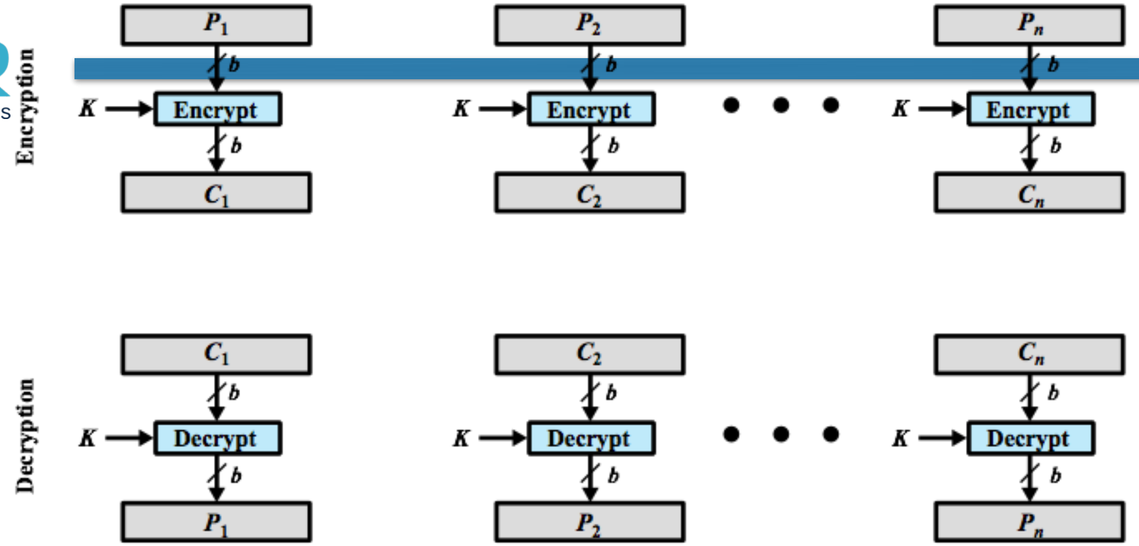
Triplo DES ou 3DES

- Aumentou a vida útil do DES
- Foi padronizado em 1985 e incorporado como parte do DES em 1999
- Envolve repetir o algoritmo DES básico três vezes, usando duas ou três chaves únicas, para obter um tamanho de chave de 112 ou 168 bits
  - **Quantas chaves diferentes o 3DES pode ter?**
- **Vantagens**: i) tamanho da chave ii) é baseado no DES, o qual já foi testado exaustivamente
- **Desvantagens**: i) é lento para implementação em software. Originalmente, foi projetado para implementação em hardware; ii) Tanto o DES, quanto o 3DES, usam um tamanho de bloco de 64 bits. Por questão tanto de eficiência quanto de segurança, um tamanho de bloco maior é desejável; iii) Outra questão importante refere-se ao nível de segurança do 3DES contra força bruta: embora a chave do 3DES seja de 168 bits, um ataque do tipo "*Meet-in-the-Middle*" reduz o custo computacional da busca para 112 bits

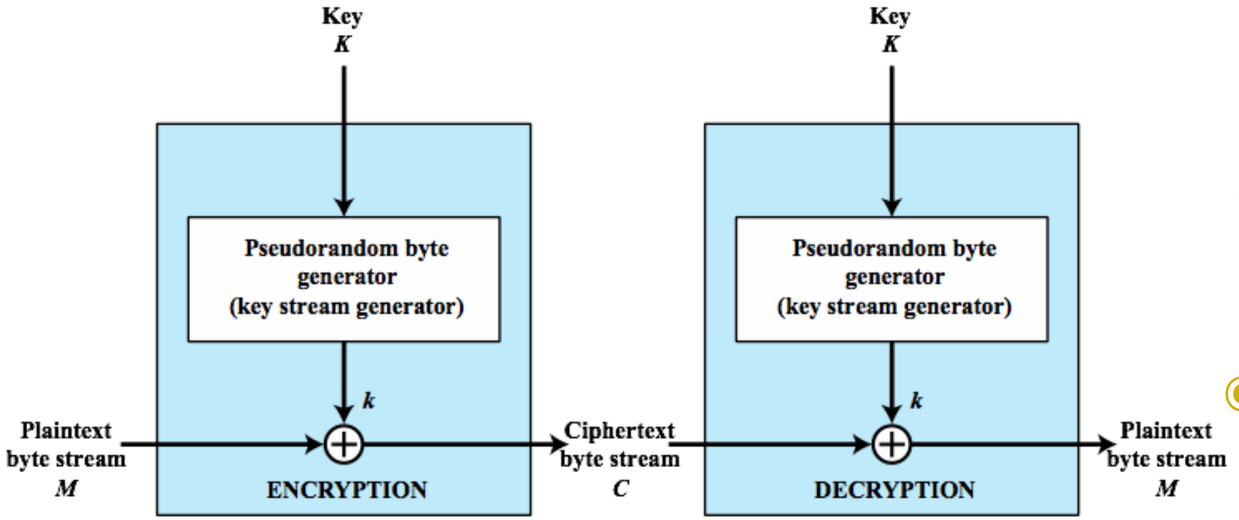
Advanced encryption standard (AES)

- Foi padronizado em 2001 pelo NIST
- O AES usa cifra de bloco simétrica com comprimento de bloco de 128 bits
- Suporte para comprimentos de chaves de 128, 192 e 256 bits
- Usado amplamente em sistemas comerciais atualmente

Cifra de bloco e Cifra de fluxo



(a) Block cipher encryption (electronic codebook mode)



(b) Stream encryption

- Uma cifra de bloco processa a entrada um bloco de elementos por vez, produzindo um bloco de saída para cada bloco de entrada
- Uma cifra de fluxo processa os elementos de entrada, continuamente, produzindo a saída de um elemento por vez, à medida que o processo avança

Cifra de bloco e Cifra de fluxo

- Para aplicações que requerem cifração/decifração de um fluxo de dados, como as que são transmitidas por um canal de comunicações de dados ou um navegador ou link da Web, uma cifra de fluxo poderia ser a melhor alternativa
- Para aplicações que lidam com blocos de dados, como transferência de arquivos, e-mail e banco de dados, cifras de bloco podem ser mais adequadas
- Todavia, qualquer um dos tipos de cifra pode ser usado em praticamente qualquer aplicação

Autenticação de mensagens e funções de Hash

Autenticação de mensagens e funções de Hash

- A cifração fornece proteção contra ataques passivos (escutas)
- Um requisito diferente é proteger contra ataques ativos (falsificação de dados e transações). A proteção contra tais ataques é conhecida como **autenticação de mensagens ou de dados**
- Autenticação de mensagens ou dados é um procedimento que permite que as partes comunicantes verifiquem se as mensagens recebidas ou armazenadas são autênticas
- **Os dois aspectos importantes são verificar se o conteúdo da mensagem não foi alterado e se a fonte é autêntica**
- Também podemos desejar verificar se uma mensagem foi transmitida no momento correto (se ela não foi artificialmente atrasada ou repetida) e a sequência em relação a outras mensagens que fluem entre duas partes
- Estas ações se referem ao atributo de **Integridade** dos dados

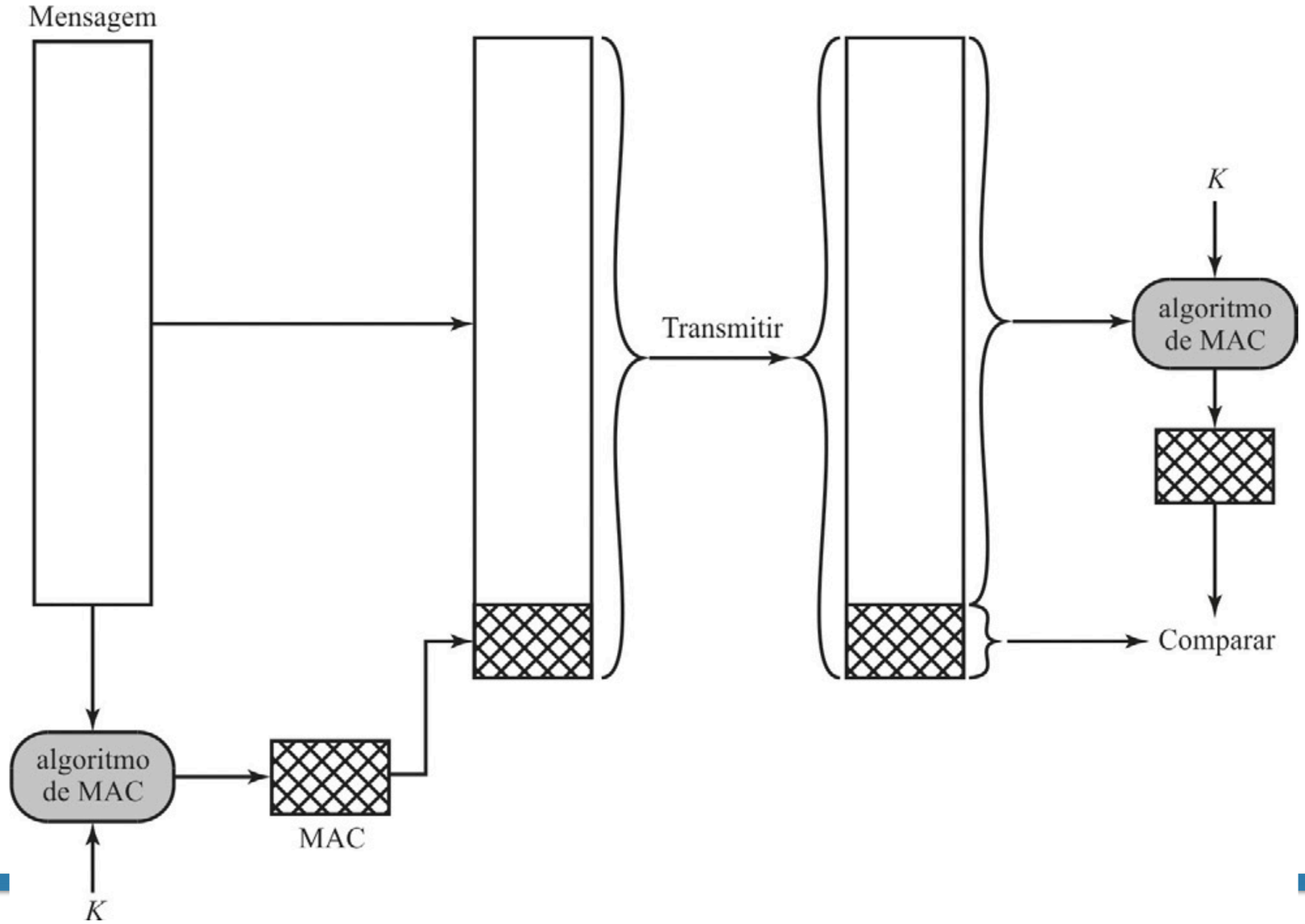
Autenticação de mensagens e funções de Hash

- É possível fazer a Autenticação de mensagem sem cifração de mensagem (i.e., sem confidencialidade)
- Um ataque como o da EFF pode, por exemplo, alterar a ordem dos blocos de textos cifrados e prejudicar a comunicação
- Três situações possíveis:
  - Há várias aplicações nas quais a mesma mensagem é transmitida a vários destinos
  - Outro cenário possível é uma troca de mensagens na qual um lado tem uma carga computacional pesada e não pode arcar com o tempo para decifrar todas as mensagens entrantes
  - A autenticação de um programa de computador em texto às claras é um serviço atraente. O programa de computador pode ser executado sem precisar ser decifrado toda vez, o que seria um desperdício de recursos do processador. Todavia, se uma tag de autenticação de mensagem for anexada ao programa, ela poderá ser verificada sempre que for exigida garantia da integridade do programa

Message authentication code (MAC)

- Uma técnica de autenticação envolve a utilização de uma chave secreta para gerar um pequeno bloco de dados, conhecido como código de autenticação de mensagem (MAC), que é anexado à mensagem
- Vários algoritmos poderiam ser usados para gerar o código MAC. A especificação NIST FIPS PUB 113 recomenda a utilização do DES

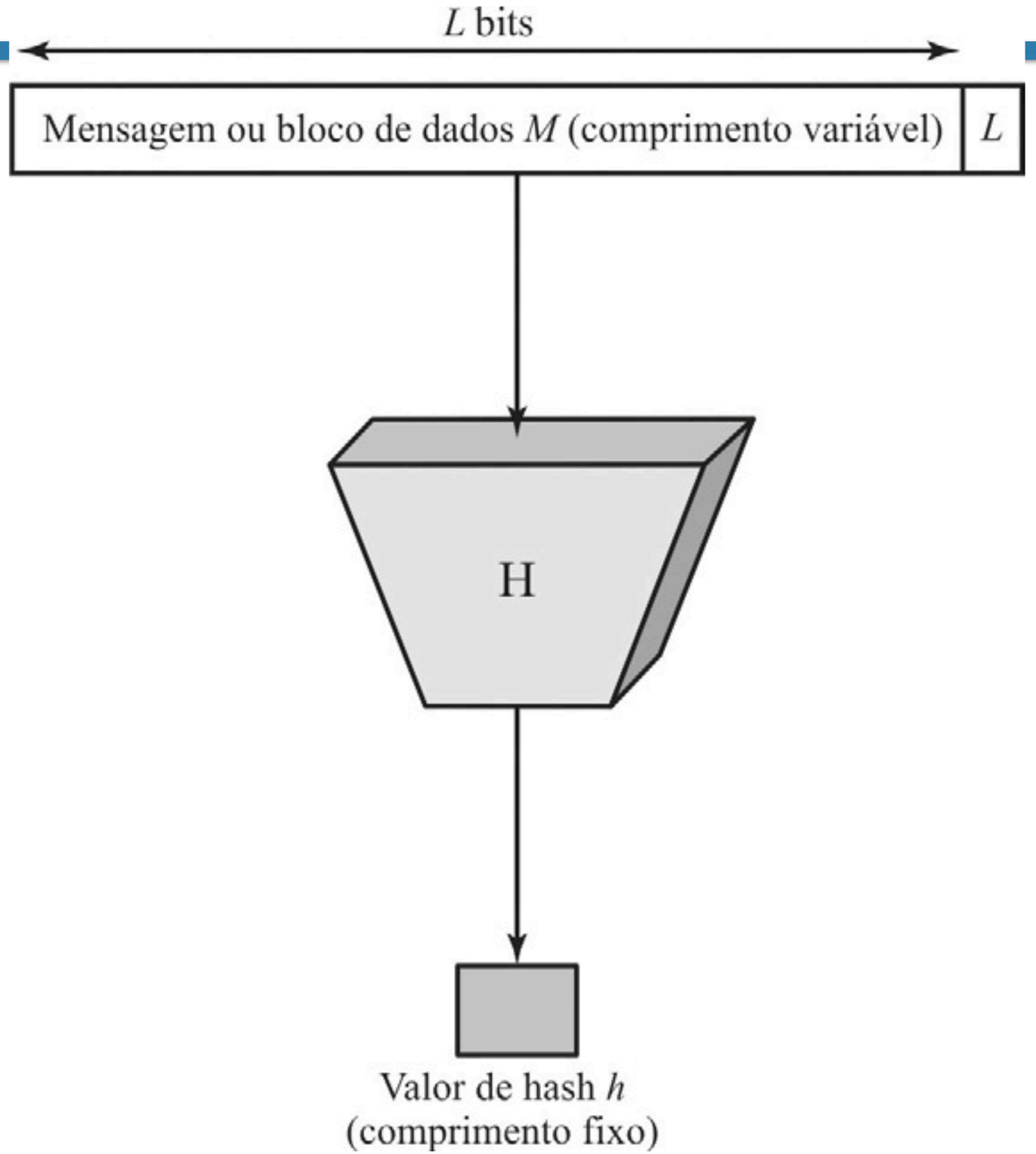
Message authentication code (MAC)



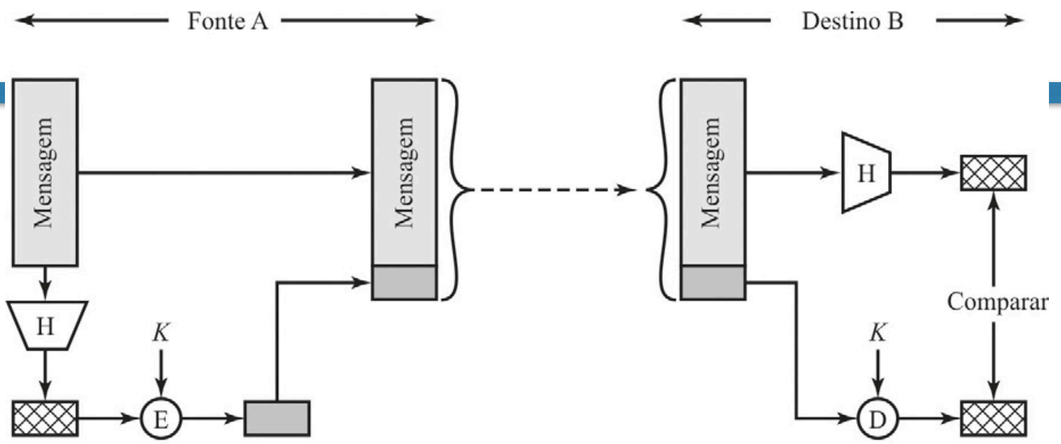
Função de hash de uma via

- Uma alternativa para o código de autenticação de mensagens é a **função de hash de uma via** (ou seja, não inversível)
- Como ocorre com o código de autenticação de mensagens, uma função de hash aceita uma mensagem M de tamanho variável como entrada e produz um resumo criptográfico de tamanho fixo da mensagem H(M) como saída
- A mensagem é preenchida até um múltiplo inteiro de algum comprimento fixo (por exemplo, 1.024 bits), e o preenchimento (também chamado de padding) inclui o valor do comprimento da mensagem original em bits
- Diferentemente do MAC, a função de hash não toma também uma chave secreta como entrada
- Para autenticar uma mensagem, o resumo da mensagem é enviado com a mensagem de modo tal que o resumo da mensagem é autêntico

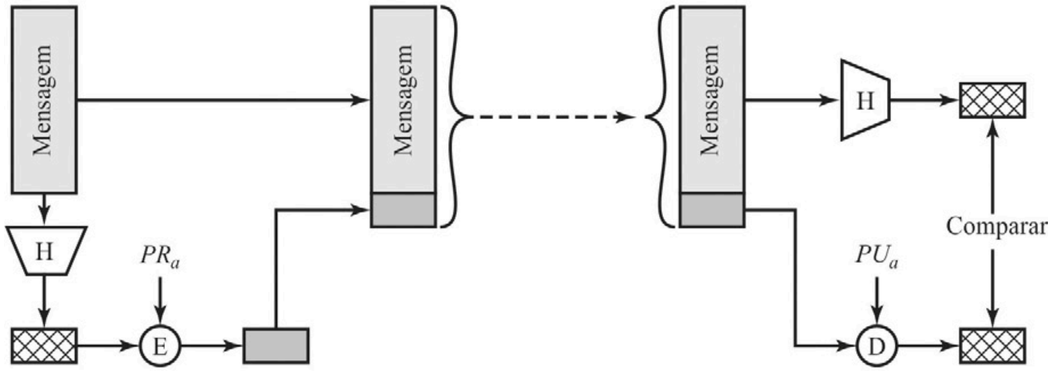
Função de hash de uma via



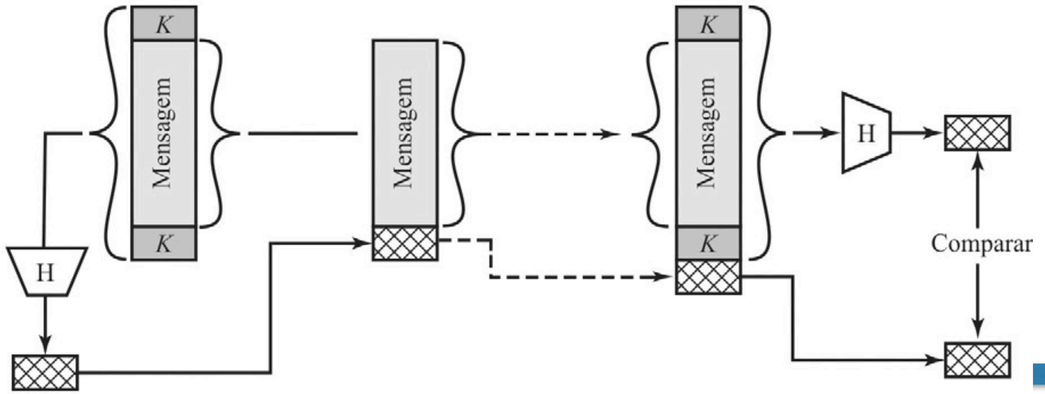
Como autenticar uma mensagem usando Função de hash de uma via?



(a) Usando cifração convencional



(b) Usando criptografia de chave pública



(c) Usando valor secreto

Requisitos de uma função de hash de uma via

- A finalidade de uma função de hash é produzir uma “impressão digital” de um arquivo, mensagem ou outro bloco de dados
- É preciso atender às seguintes propriedades

1. H pode ser aplicada a um bloco de dados de qualquer tamanho.
2. H produz uma saída de comprimento fixo.
3. $H(x)$ é relativamente fácil de computar para qualquer x dado, tornando práticas implementações em hardware e em software.
4. Para qualquer código dado h, é inviável em termos computacionais achar x tal que $H(x) = h$. Uma função de hash com essa propriedade é denominada **via** ou **resistente à pré-imagem.**[10]
5. Para qualquer bloco dado x, é inviável em termos computacionais achar $y \neq x$ tal que $H(y) = H(x)$. Uma função de hash com essa propriedade é denominada **resistente à segunda pré-imagem,** às vezes denominada **resistente a colisão fraca**.
6. É inviável, em termos computacionais, achar qualquer par (x, y) tal que $H(x) = H(y)$. Uma função de hash com essa propriedade é denominada **resistente a colisão**, às vezes, **resistente a colisão forte.**

As três primeiras propriedades são requisitos para a aplicação prática de uma função de hash para autenticação de mensagens.

Algoritmos de função de hash seguros

- Ao longo dos anos, o NIST tem padronizados muitos algoritmos para geração de funções hash (*Secure Hash Algorithm* (SHA — algoritmo de hash seguro))
- Atualmente, o SHA possui várias versões em uso: SHA-256, SHA-384 e SHA-512, com comprimentos de valor de hash de 256, 384 e 512 bits, respectivamente
- O SHA-1 foi descontinuado. Produzia um valor de hash de 160 bits
- Outros exemplos de uso de funções hash
  - Senhas: hash de senhas
  - Detecção de intrusão: hash de arquivos

Criptografia de chave pública ou criptografia assimétrica

Criptografia de chave pública ou assimétrica

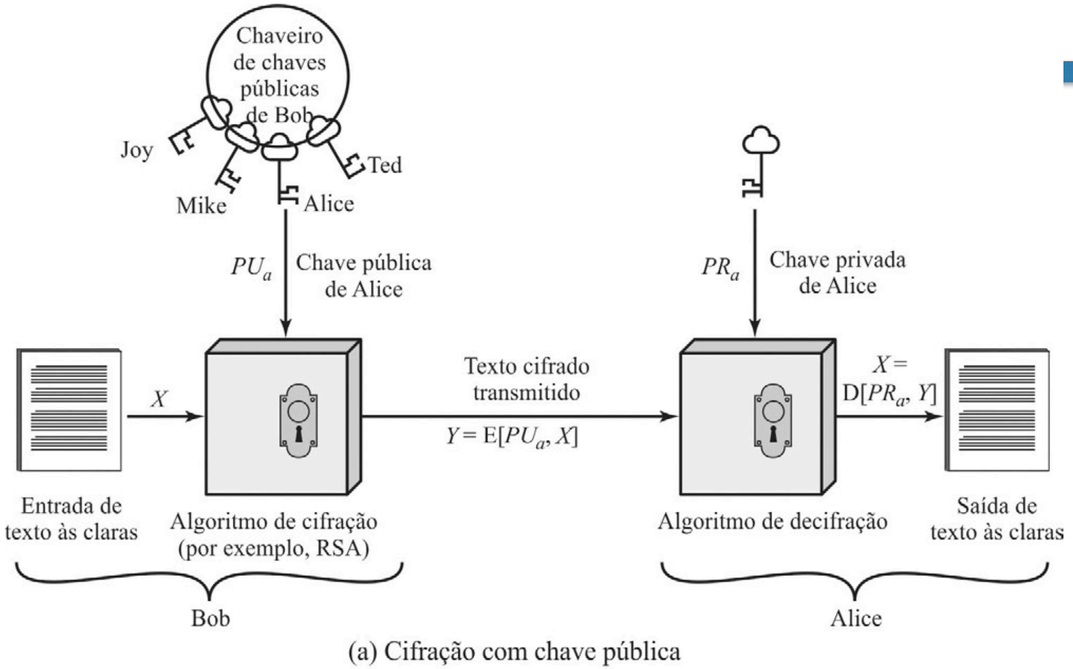
- É utilizada para autenticação de mensagens e distribuição de chaves
- Proposta por Diffie e Hellman em 1976
- São baseados em funções matemáticas em vez de simples operações sobre sequências de bits
- Utiliza duas chaves separadas, em vez de uma chave como na criptografia simétrica

Criptografia de chave pública ou assimétrica

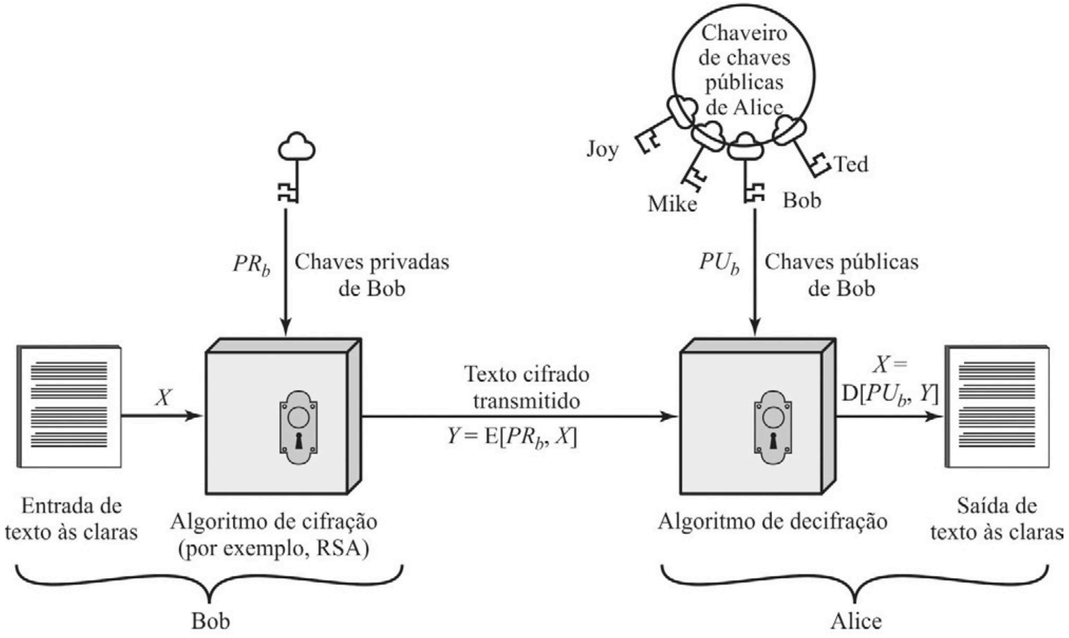
- Componentes de uma técnica de criptografia assimétrica:
  - **Texto as claras**: é a mensagem ou dados originais dado como entrada do algoritmo de cifração
  - **Algoritmo de cifração**: O algoritmo executa várias substituições e transformações no texto as claras
  - **Chaves pública e privada**: É um par de chaves que foi selecionado de modo que, se uma é usada para cifrar, a outra é usada para decifrar. As transformações exatas executadas pelo algoritmo criptográfico dependem da chave pública ou privada que é passada como entrada
  - **Texto cifrado**: mensagem embaralhada produzida como saída
  - **Algoritmo de decifração**: Esse algoritmo aceita o texto cifrado e a chave correspondente, e produz o texto às claras original

Etapas essenciais da técnica

1. Cada usuário gera um par de chaves a ser usado para a cifração e a decifração de mensagens
2. Cada usuário coloca uma das duas chaves em um registro público ou outro arquivo acessível. Essa é a chave pública. A chave associada é mantida privada. Cada usuário mantém uma coleção de chaves públicas obtidas de outros
3. Se Bob desejar enviar uma mensagem privada a Alice, ele cifra a mensagem usando a chave pública de Alice
4. Quando Alice recebe a mensagem, ela a decifra usando sua chave privada. Nenhum outro destinatário pode decifrar a mensagem porque somente Alice sabe qual é a sua chave privada



(a) Cifração com chave pública



(b) Cifração com chave privada

promove confidencialidade: somente o destinatário-alvo é capaz de decifrar o texto cifrado, pois só ele tem a chave privada

promove autenticação e/ou integridade de dados

1. É computacionalmente fácil para uma entidade B gerar um par (chave pública PU_b, chave privada PR_b).
2. É computacionalmente fácil para um remetente A, que conheça a chave pública e a mensagem a ser cifrada, M, gerar o texto cifrado correspondente:

$$C = E(PU_b, M)$$

3. É computacionalmente fácil para o destinatário B decifrar o texto cifrado resultante usando a chave privada para recuperar a mensagem original:

$$M = D(PR_b, C) = D[PR_b, E(PU_b, M)]$$

4. É computacionalmente inexequível para um oponente que conheça a chave pública, PU_b, determinar a chave privada, PR_b.
5. É computacionalmente inexequível para um oponente que conheça a chave pública, PU_b e um texto cifrado, C, recuperar a mensagem original, M.

Podemos ainda adicionar um sexto requisito que, embora útil, não é necessário para todas as aplicações de chave pública:

6. Qualquer das duas chaves relacionadas pode ser usada para a cifração, sendo a outra usada para a decifração.

$$M = D[PU_b, E(PR_b, M)] = D[PR_b, E(PU_b, M)]$$

Requisitos para criptografia de chave pública

UFPB

Algoritmos criptográficos assimétricos

1. **RSA**
   - foi desenvolvido em 1977 por Ron Rivest, Adi Shamir e Len Adleman no MIT
   - é o algoritmo de criptografia assimétrica mais utilizado atualmente
   - deve-se usar um tamanho de chave de pelo menos 2048 ou 3072 bits
2. **Acordo de chaves de Diffie-Hellman**
   - A finalidade do algoritmo é permitir que dois usuários cheguem a um acordo seguro sobre um segredo compartilhado que pode ser usado como chave secreta para subsequente aplicação de criptografia simétrica sobre mensagens
   - O algoritmo em si é limitado à troca das chaves

Algoritmos criptográficos assimétricos

3. ***Digital signature standard*** **(DSS)**
   - O DSS usa um algoritmo projetado para prover somente a função assinatura digital
   - Diferentemente do RSA, ele não pode ser usado para cifração ou troca de chaves
4. **Criptografia de curvas elípticas (ECC -** ***Elliptic Curve Cryptography*****)**
   - Alternativa ao RSA, pois prover a mesma segurança com um tamanho de chave menor
   - O nível de confiança na ECC ainda não é tão alto quanto no RSA

Aplicações para criptossistemas de chave pública

Tabela 2.3 Aplicações para criptossistemas de chave pública

Algoritmo	Assinatura digital	Distribuição de chave simétrica	Cifração de chaves secretas
RSA	Sim	Sim	Sim
Diffie-Hellman	Não	Sim	Não
DSS	Sim	Não	Não
Curvas elípticas	Sim	Sim	Sim

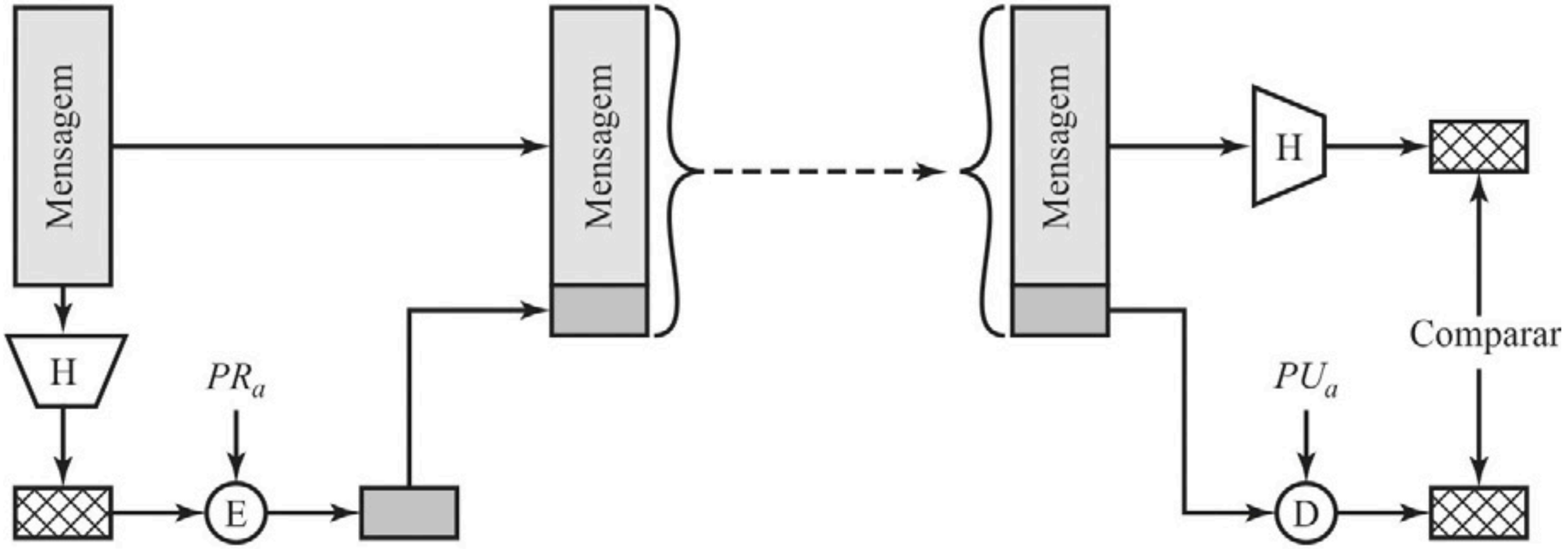
Assinaturas Digitais e Gerenciamento de Chaves

Assinaturas Digitais

- Algoritmos de chave pública são usados principalmente em duas categorias de aplicações: assinaturas digitais e várias técnicas que têm a ver com gerenciamento e distribuição de chaves
- Três aspectos em relação à utilização de criptografia de chave pública no contexto de gerenciamento e à distribuição de chaves:
  - A distribuição segura de chaves públicas
  - A utilização de criptografia de chave pública para distribuir chaves secretas
  - A utilização de criptografia de chave pública para criar chaves temporárias para a cifração de mensagens

Assinaturas Digitais

- A criptografia de chave pública pode ser usada para autenticação. No exemplo, foi criada uma assinatura digital do usuário transmissor
- É importante enfatizar que a assinatura digital não provê confidencialidade. Isto é, a mensagem que está sendo enviada está a salvo de alteração, mas não de escuta por terceiros.

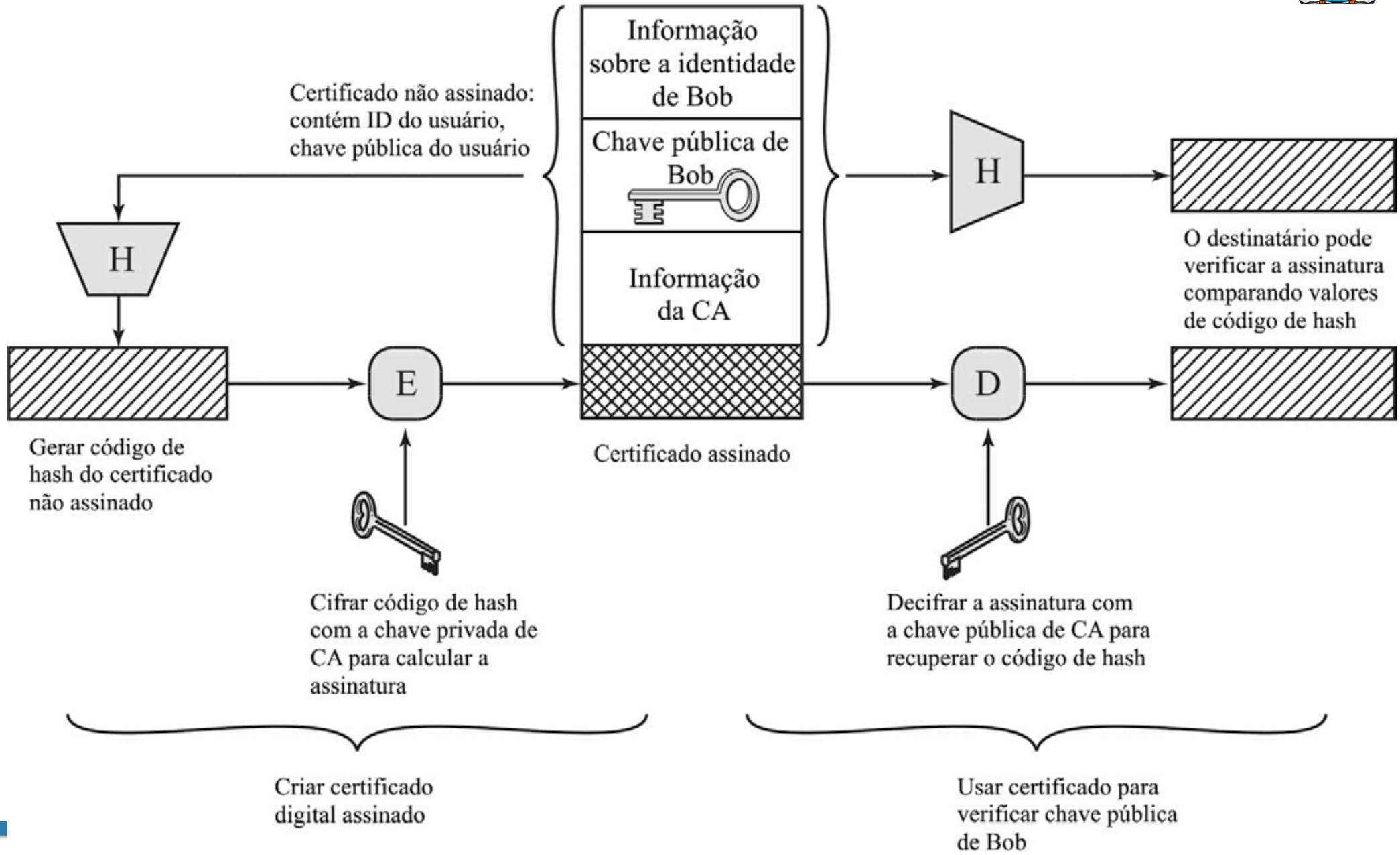


(b) Usando criptografia de chave pública

Certificados de chave pública

- É importante ter um entidade certificadora (CA - *Certification Authority*) de chaves públicas
- Um certificado de chave pública consiste em uma chave pública mais um ID de usuário do proprietário da chave, e o bloco inteiro assinado por uma terceira entidade confiável
- O padrão X.509 é utilizado amplamente para certificação de chave pública
- Certificados X.509 são usados na maioria das aplicações de segurança de rede, incluindo IP *Security* (IPsec), *Transport Layer Security* (TLS), *Secure Shell* (SSH) e *Secure/Multipurpose Internet Mail Extension* (S/MIME)

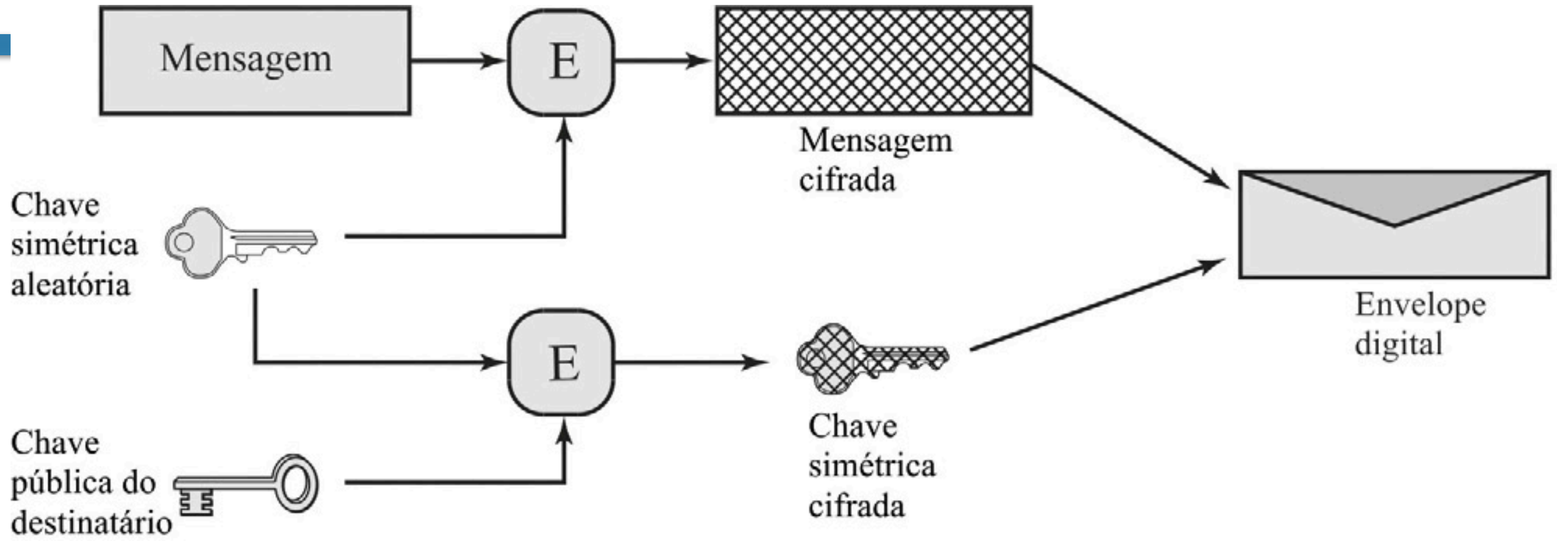
Certificados de chave pública



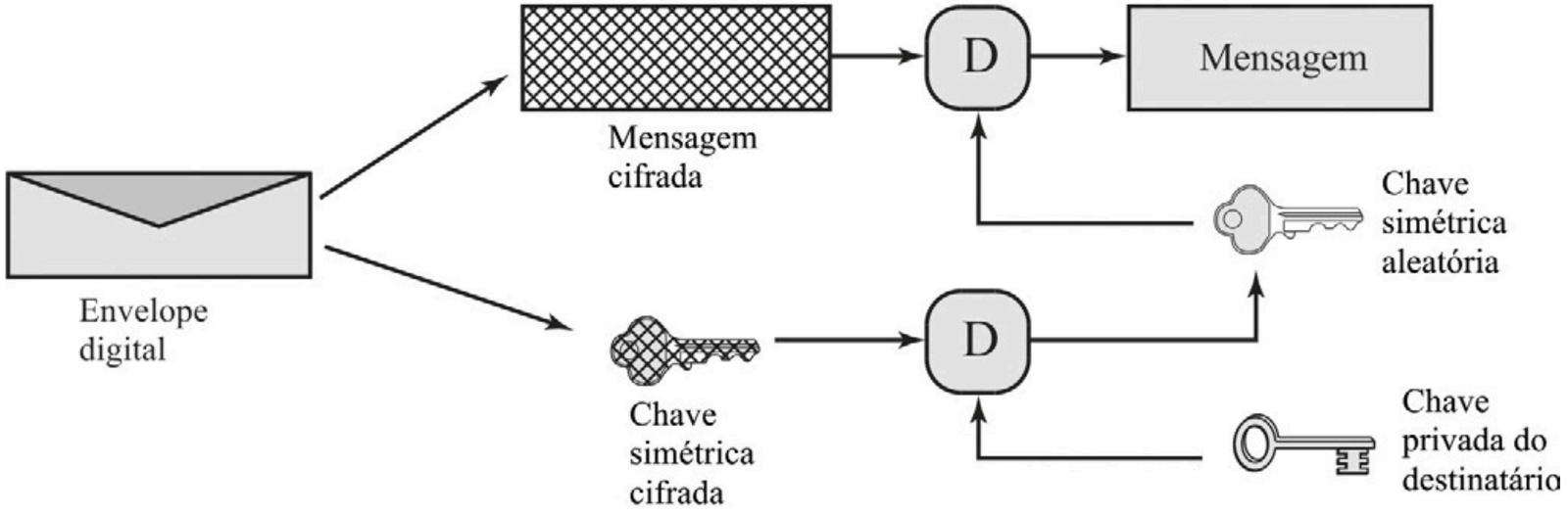
Envelope Digital

- Outra aplicação na qual a criptografia de chave pública é usada para proteger uma chave simétrica é o envelope digital
- Envelope digital pode ser usado para proteger uma mensagem sem antes exigir que o remetente e o destinatário tenham a mesma chave secreta
- Neste caso, a chave secreta aleatória é criada e usada uma única vez

Envelope Digital



(a) Criação de um envelope digital



(b) Abertura de um envelope digital

Obrigado!

Perguntas e encaminhamentos

Iguatemi E. Fonseca

iguatemi@ci.ufpb.br