Criptografia

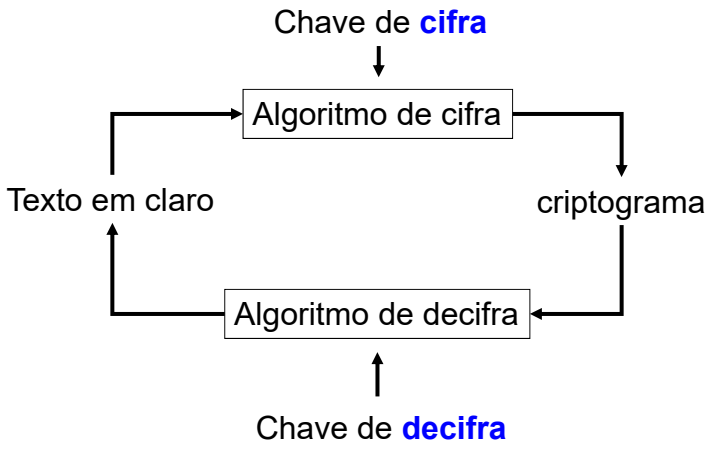
Nuno Neves
Departamento de Informática
Faculdade de Ciências da Universidade de Lisboa

Sumário

Introdução

- Criptografia
  - Krypthós (oculto) + graph (escrever)
  - ☹ o uso de criptografia revela-a
    - fornece indícios de que a informação é sensível
    - pode ser ilegal
- Esteganografia
  - Conteúdo sensível é ocultado dentro de outro conteúdo
  - Exemplos:
    - escrita com tinta invisível
    - ocultar conteúdos dentro de imagens nos bits menos significativos de cada pixel
- Criptanálise
  - Arte ou ciência de violar informação criptografada ou sistemas criptográficos
- Criptologia
  - Estudo de criptografia e criptanálise

Criptografia



Criptanálise

- Objectivos:
  - obter texto original
  - obter chave de cifra
  - obter algoritmo de cifra
- Algumas técnicas
  - ataques usando apenas o criptograma (ciphertext-only attacks)
  - ataques com conhecimento de parte do texto original (known-plaintext attacks)
  - ataques com texto original escolhido (chosen-plaintext attacks)
    - ataques com texto original escolhido de forma adaptativa (adaptive chosen-plaintext attacks)
  - ataques com criptogramas escolhidos (chosen-ciphertext attacks)
  - ataques de aniversário (birthday attacks)

Evolução da tecnologia de cifra

- Primeiras cifras
  - a sua segurança era baseada no secretismo do algoritmo
  - Espartanos
    - o pergaminho só poderia ser lido se fosse enrolado num bastão com o mesmo diâmetro

  - Cifra de César
    - Substitui uma letra pela $k^{ésima}$ letra seguinte no alfabeto, MOD 26
    - Mecanismo:
      - $E_k(m) = (m + k) \bmod 26$ (função de cifrar)
      - $D_k(c) = (26 + c - k) \bmod 26$ (função de decifrar)
    - Exemplo:
      - $k = 2$
      - Texto em claro: seguranca
      - Texto cifrado: ugixtcpec

Cifras de transposição (ou permutação)

- Baralham (i.e., trocam a ordem) os caracteres do texto original
- Exemplos:
  - Permutações fixas em blocos com um número constante de caracteres
    - permutação: 45231
    - criptograma: raifc
    - Qual é o texto original ?
  - Blocos verticais de dimensão fixa
    - blocos verticais de 5 caracteres:

eaeo …
loms
esqo
saun
nbeh

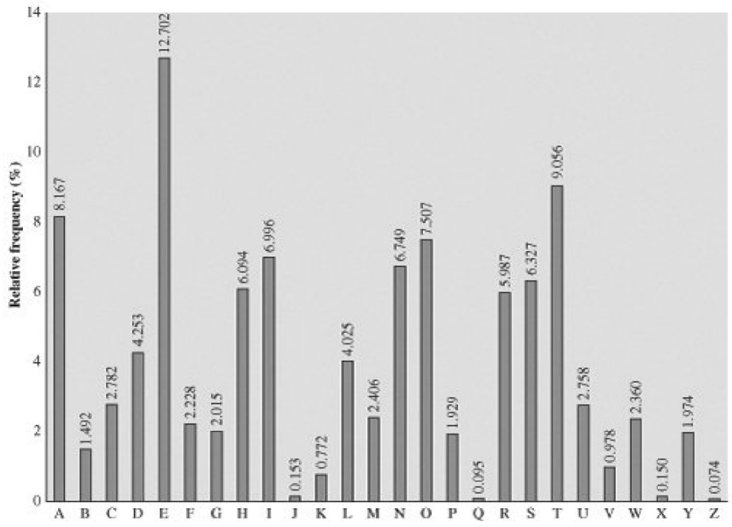
Text-cifrado

eaeo loms esqo saun nbeh

Cifras de substituição

- Substituem os caracteres do alfabeto usado no texto original por caracteres de um alfabeto de substituição
- **Mono-alfabéticas**
  - usam apenas um alfabeto de substituição
  - um carácter do alfabeto original é substituído sempre pelo mesmo carácter
  - exemplo: cifra de César
  - Criptoanálise
    - força bruta experimentando todas as combinações
    - *padrões estatísticos* dos caracteres usados no texto original (prox. slide)
    - ataques com texto original escolhido
- **Poli-alfabéticas**
  - aplicação sucessiva e cíclica de várias cifras mono-alfabéticas
  - exemplo: Cifra de Vigenère; Máquinas de rotor (slides seguintes)

Exemplo de frequência de letras no Inglês



Cifra Vigenère

- Mecanismo
  - chave K é um conjunto de caracteres (e.g., uma palavra)
  - repete-se a chave em sequência até que a chave seja do tamanho do texto a ser cifrado
  - aplica-se a letra do texto em claro a substituição que corresponde à letra correspondente da chave
  - se a chave tem uma letra apenas, temos uma cifra monoalfabética
- Exemplo de uso
  - K = poema

```
Chave              poemapoemapoe
Texto em claro     elesnaosabemq
Texto cifrado      tzienpcwmbtau
```

- Criptanálise
  - determinar a dimensão da chave N -> criptanálise de N cifras monoalfabéticas
  - técnicas estatísticas para determinar N: teste de Kasiski e índice de coincidência

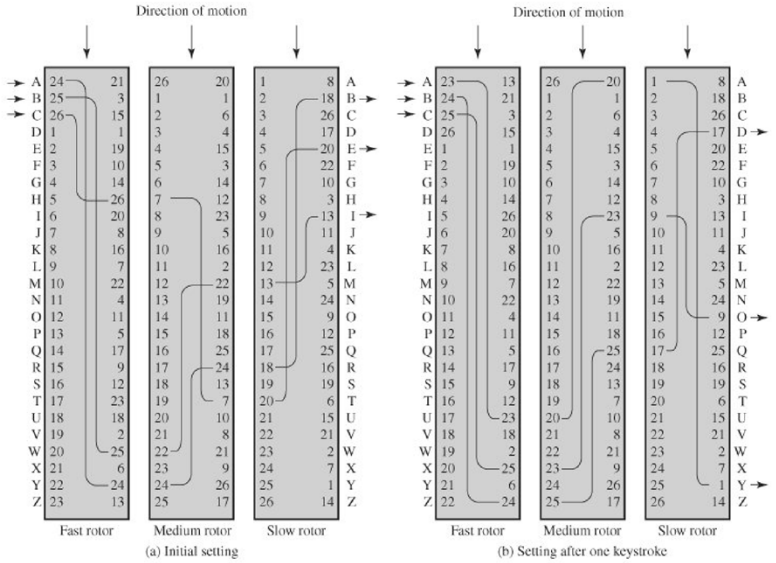
Tabela de Vigenère

Key \ Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Máquinas de Rotor

- Máquinas usadas para cifrar e decifrar mensagens
- Antes dos computadores modernos, eram muito usadas
  - na Segunda Guerra Mundial:
    - Enigma (Alemanha),
    - Hagelin (Aliados),
    - Purple (Japão)
- Formas de concretização variam (ex. uso de vários cilindros), mas em geral baseavam-se no uso de várias cifras de substituição, o que tornava a criptoanálise extremamente complexa

Máquina de Rotor com 3 Rotores



Conceitos teóricos

- Cifra perfeita
  - uma cifra diz-se **perfeita** quando, dado um criptograma *c*, a probabilidade de ele corresponder a um dado texto original *m* e de ter sido gerado com uma dada chave *k* é igual à probabilidade de ocorrência do texto *m*
  - o cardinal do espaço de chaves tem de ser igual ou superior ao cardinal do espaço de textos em claro
  - Cifra de Vernam (serviu de base para *one time pad*)

  - Dificuldades
    - para cada texto tem de ser usada uma chave diferente
    - o comprimento das chaves tem de ser igual ou superior ao dos textos
    - as chaves não são memorizáveis
    - pré-distribuição de chaves de grande dimensão
    - não faz sentido usar para cifrar dados armazenados

Conceitos teóricos

- **Cifras seguras**
  - uma cifra diz-se segura se cumprir o objetivo para que é usada
    - a cifra não permite a sua criptanálise em **tempo útil** e admitindo um **investimento** tendo em conta a relação **custo-benefício**

- Critérios para avaliar a qualidade das cifras [Claude Shannon,1949]
  1. Quantidade de secretismo oferecida
     - tempo mínimo de segurança do criptograma
  2. Dimensão das chaves
     - cifra de Vernam – gestão de chaves ☹
  3. Simplicidade de realização e uso
  4. Propagação de erros
  5. Dimensão do criptograma
     - o tamanho do texto cifrado não deve ser maior que o do texto em claro, uma vez que tem impacto no custo de armazenamento ou transmissão

Conceitos teóricos

- Boas práticas:
  - Criptanalista conhece o algoritmo de cifra e as suas fragilidades
    - Segurança baseia-se no desconhecimento da chave
    - Tem de ser baseado em matemática sólida
    - Tem de ter sido analisado por vários especialistas
    - Tem de ter passado no teste do tempo
  - Criptanalista pode capturar todos os criptogramas
  - Criptanalista conhece partes do texto original

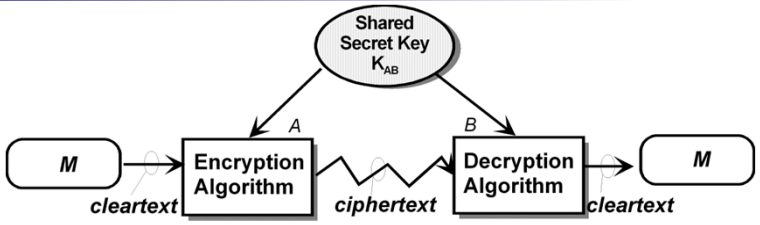
Sumário

- Introdução
- Criptografia e criptanálise
- Evolução da tecnologia de cifra
- Tipos de cifra
  - Cifras de transposição
  - Cifras de substituição
    - Monoalfabéticas
    - Polialfabéticas
- Conceitos teóricos
- **Cifras modernas**
  - **Cifras simétricas**
  - Cifras assimétricas
  - Cifras por blocos
    - Modos de cifra
    - Reforço de segurança
  - Funções de síntese
  - Autenticação
    - MACs
    - Assinaturas digitais

Cifras modernas

- Modo de operação
  - Cifras por blocos
  - Cifras contínuas
- Tipo de chave
  - Cifras simétricas
    - Chave secreta
    - Confidencialidade
    - Eficientes
    - N utilizadores e distribuição segura de chaves ☹
  - Cifras assimétricas
    - Par de chaves
    - Confidencialidade, autenticidade
    - Não eficientes
    - N utilizadores ☺
    - Distribuição de chaves públicas
  - Cifra híbrida
    - Cifra com chave simétrica
    - Distribuição de chave simétrica com cifras assimétricas
    - **Porquê?**

Criptografia Simétrica



- Também denominada de chave partilhada ou de chave secreta
  - chave de cifrar e decifrar iguais
  - bastante rápida
- Propriedade fundamental: D(K, E(K, m)) = m
- Cifras simétricas por blocos
  - *Data Encryption Standard* (DES) (1977); *Triple-DES*;
  - International Data Encryption Algorithm (IDEA);
  - *Advanced Encryption Standard* (AES) (2000)
- Cifras simétricas contínuas
  - A5 (usado no GSM)
  - RC4

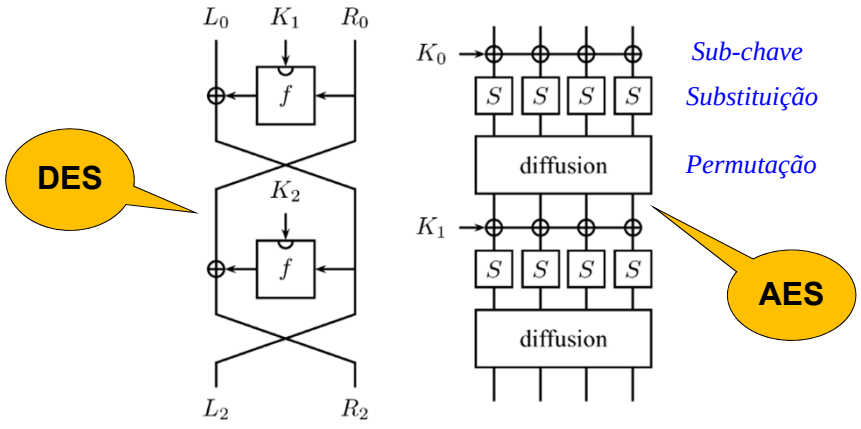
Criptografia Simétrica – ponto de situação

- O **Data Encryption Standard (DES)** foi o primeiro algoritmo estandardizado e usado massivamente para melhorar a segurança de sistemas computacionais (1975)
- O **DES** resistiu (na prática) aos diversos ataques de criptanálise, acabando por ser atacado devido ao seu pequeno tamanho de chave (56 bits)
  - muitos desafios foram lançados para quebrar mensagens específicas cifradas com o DES
  - a maioria foi resolvida usando computação paralela e/ou distribuída
- O NIST em 2001 seleccionou o algoritmo **Rijndael** como seu **Advanced Encryption Standard (AES)**, o sucessor do DES
  - desenhado para resistir a ataques bem sucedidos ao DES

AES - Advanced Encryption Standard

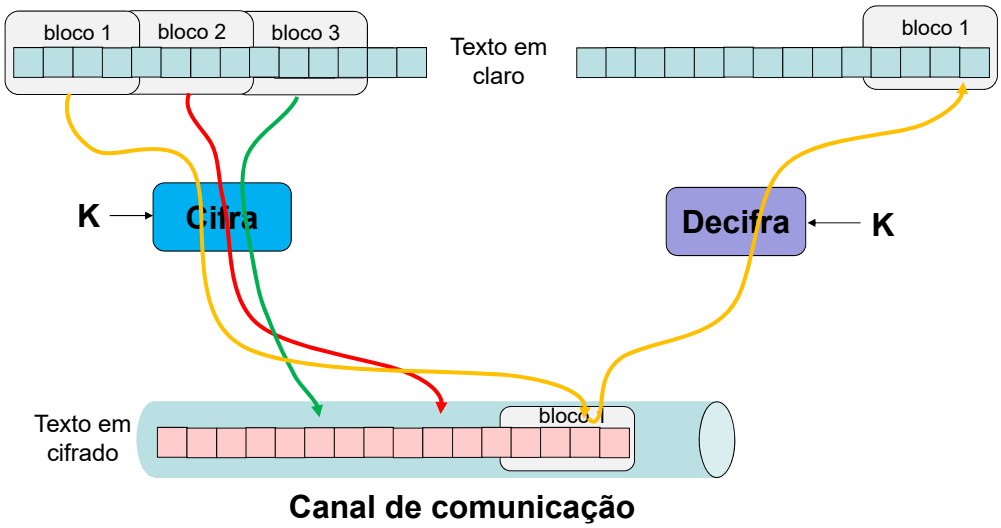
- Histórico:
  - Novo padrão do NIST para substituir o DES
  - Processo de selecção público (iniciado em 1997) onde se escolheu um algoritmo de entre vários candidatos
  - O escolhido foi o Rijndael
- Princípios:
  - Recebe como entrada blocos de 128 bits de texto em claro
  - As chaves podem ter 128, 192, 256 bits (quanto maior, mais seguro)
  - Produz blocos de 128 bits de texto cifrado
  - Funciona iterativamente
    - cada bloco é dividido em 4 grupos de 4 bytes
    - um bloco inteiro é modificado em cada iteração (no DES é só metade)
  - Rápido e eficiente em CPUs pequenos e grandes

AES - Advanced Encryption Standard



Estrutura de Feistel Rede SP

Sumário: Cifra simétrica de blocos



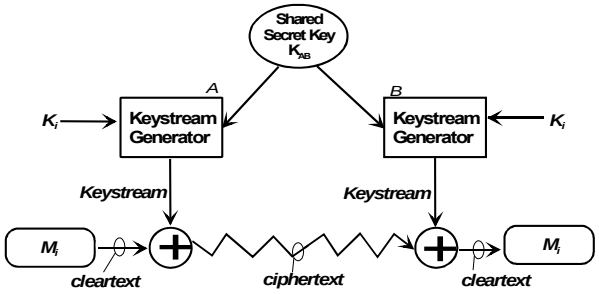
Cifras simétricas contínuas (Stream Cipher)

- Poder do XOR (OU Exclusivo)
  - operação binária com duas entradas onde a saída é 1 se e somente se uma e apenas uma das entradas é igual a 1

Input		Output
A	B	
0	0	0
0	1	1
1	0	1
1	1	0

- Repetição de dois XORs
  - ((B XOR A) XOR A) = B
- Se A é um bit da **chave de fluxo**, que é do ponto de vista prático aleatória, então temos uma cifra muito eficiente e segura

Cifras simétricas contínuas



- PRINCÍPIO:
  - cifra de fluxo processa um bit/byte de cada vez através da operação XOR
  - a partir de **chave partilhada** é produzida uma sequência infinita de bits/bytes aleatórios a que se chama a **chave de fluxo** ou sequência (*keystream*)
  - a chave de fluxo é usada uma única vez e portanto é muito difícil a criptoanálise
- CIFRAR:
  - a chave de fluxo é XOR ao correr do fluxo de texto em claro, bit a bit (ou byte a byte)
- DECIFRAR:
  - o fluxo cifrado é XOR com a mesma chave de fluxo, o que retorna o fluxo original

Cifras simétricas contínuas

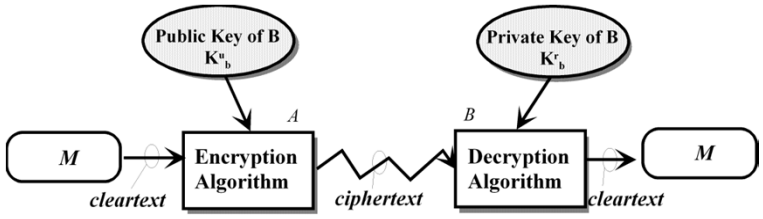
- Requisitos de robustez:
  - Secretismo, aleatoriedade e uso único da **chave de fluxo**
  - A chave de fluxo precisa ser distribuída nas duas pontas do canal
- Uso em sistemas reais:
  - Em comunicação, a *chave de fluxo* é uma **sequência pseudo-aleatória** produzida em tempo-real à velocidade do fluxo de texto, por uma caixa-preta;
  - A chave de fluxo é gerada nos dois extremos em simultâneo (as duas caixas pretas são sincronizadas)
  - A chave de fluxo é parametrizada por uma chave mestra
  - É susceptível a erros de bits, que podem dessincronizar o fluxo
  - Exemplo de algoritmo: RC4 (usado no SSL e no WAP)

Criptografia Simétrica – vantagens/desvantagens

- Eficientes ☺

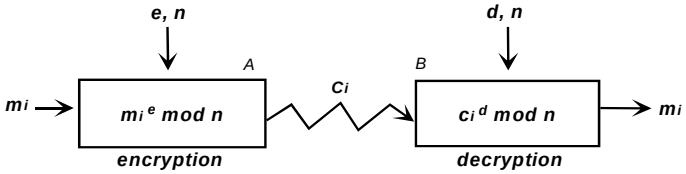
- Chave secreta
  - se perdida ou revelada em qualquer ponta, o canal é comprometido
- Distribuição de chaves
  - o problema do ovo e da galinha: *"como distribuir a chave para ter canais seguros sem ter canais seguros?"*
  - e se as chaves precisarem de ser mudadas frequentemente
- Gestão de chaves
  - grande escala ☹
  - comunicação arbitrária entre 10 participantes requer 45 chaves
  - 100 participantes -> quase 5000
  - *(n(n-1)/2)* chaves são requeridas para *n* participantes

Criptografia Assimétrica



- Também chamada de cifra de chave pública
  - cifra com chave pública K_u e decifra com chave privada K_r
  - em geral é muito mais lenta que a criptografia simétrica
- PRINCÍPIO
  - usam problemas matemáticos, para os quais não existe solução em tempo polinomial, aplicados a grande números – factorização e o cálculo de logaritmos discretos
- PROPRIEDADES:
  - $D(K_r,E(K_u,m))=m$ e $E(K_u,D(K_r,m))=m$
- Exemplos de Algoritmos:
  - Rivest-Shamir-Adleman (RSA) (1978), ElGamal;
  - Diffie-Hellman, para calcular um número secreto partilhado (1976);

Criptografia Assimétrica - RSA



- Foi publicado em 1977 por três investigadores do MIT: Rivest, Shamir e Adleman (RSA)
- Pode ser usado tanto para cifrar quanto para assinar

- Texto em claro é dividido em blocos que são tratados como um número
- Exponenciação com a chave pública para obter o criptograma
- Exponenciação com a chave privada para obter o texto em claro

Criptografia Assimétrica - RSA

- Gerar Chaves:
  - Escolhe dois números primos grandes p, q
  - Considere $n = pq$ e $z = \phi(n) = (p–1)(q–1)$
  - Escolha $e < n$ tal que e é primo relativo (não tem factores em comum) de z
  - Calcula d tal que $ed \bmod z = 1$
  - chave pública: $K_U = (e, n)$; chave privada: $K_r = (d, n)$
- Cifrar:
  - $E(K_U,m) = m^e \bmod n = c$
- Decifrar:
  - $D(K_r,c) = c^d \bmod n = m$

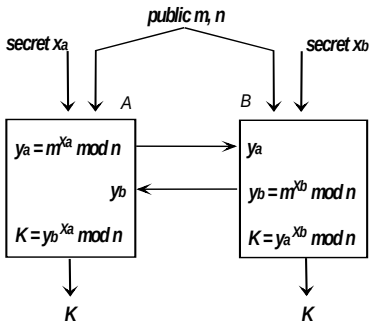
É mais lento à medida que d e e crescem:
(d é geralmente grande enquanto e é pequeno)

Criptografia Assimétrica - RSA

- Criptanálise:
  1. Procura de chaves "à bruta"
     - Inexequível se usarmos chaves grandes (≥ 1024 bits)
  2. Ataques matemáticos
     - *d* é fácil de calcular a partir de *e* se forem conhecidos *p* e *q* -> factorização de nºs grandes
     - Determinar *m* a partir de *c*, *e* e *n* -> Função inversa da exponenciação modular: logaritmo modular
     - ainda seguro com chaves ≥ 1024 bits
  3. Ataques temporais (*timing attacks*) na execução da operação de decifração
     - consegue estimar *d* pelo tempo que demora uma decifração

Criptografia Assimétrica - Diffie-Hellman

- PRINCÍPIO:
  - é baseado em uma *one-way function* (função irreversível) e na dificuldade de se obterem logaritmos discretos
- OBJECTIVO:
  - obter um número secreto K, compartilhado entre A e B, sem o comunicar em claro
- OPERAÇÃO:
  - escolher dois números primos *m* e *n* públicos (n grande)
  - A gera um número aleatório x_a
  - A calcula $y_a = m^{x_a} \bmod n$
  - B gera um número aleatório x_b
  - B calcula $y_b = m^{x_b} \bmod n$
  - y_a e y_b são tornados públicos
  - Cada um calcula *K* localmente
  - $K = y_b^{x_a} \bmod n = y_a^{x_b} \bmod n = m^{x_a x_b} \bmod n$



Segurança do Diffie-Hellman

- A segurança do Diffie-Hellman é baseada na dificuldade em se resolver o seguinte problema:
  - Dados os elementos n, m e os valores m^x e m^y, qual o valor de m^{xy}?
- Isto é equivalente ao calculo de logaritmos discretos:
  - $x = log_m (m^x)$
  - $y = log_m (m^y)$

  (seria relativamente simples se não estivessemos a falar de aritimética modular – lembrem-se dos *"mod n"* nas fórmulas)

Criptografia Assimétrica - vantagens/desvantagens

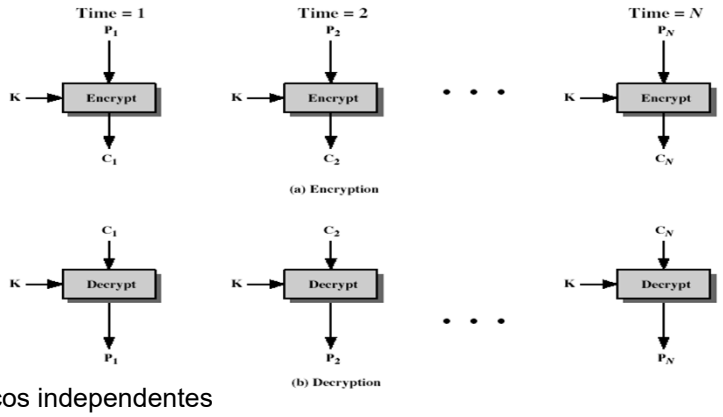
- Eficiência ☹
- Escala ☺
- Distribuição de chaves (públicas) ☹
  - autenticidade das chave públicas
  - Como saber se afirmações do tipo "12DH457B6A9 é chave pública do Pedrinho" são verdadeiras?
  - Um chave pública a ser enviada pode ser interceptada e substituída…
  - Ou, se uma base de dados de chaves públicas (PKI ou CA) é comprometida, qualquer chave armazenada pode ser substituída por uma chave falsa criada pelo atacante
  - A definição de **autoridades de certificação** (alguém que certifique a autenticidade das chaves) é necessária

MODOS DE CIFRA PARA ALGORITMOS SIMÉTRICOS DE BLOCOS

Como pegar num algoritmo de cifra básico e dar-lhe várias utilizações?

ECB: Electronic Code Book

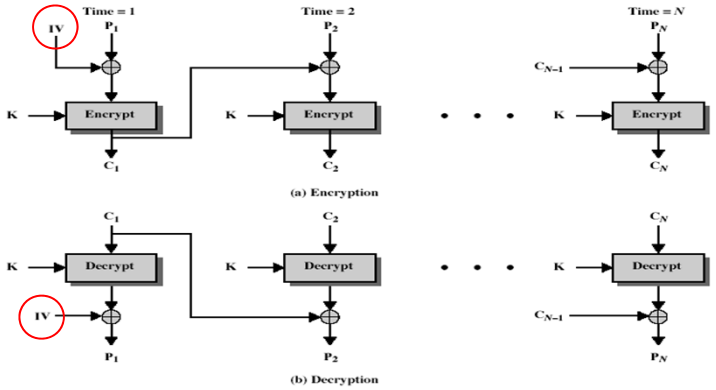
ECB:
Electronic CodeBook



- Cifra por blocos independentes
- Fraquezas
  - reprodução de padrões de texto original – dois blocos iguais produzem o mesmo criptograma
  - vulnerável a ataques de reordenação ou *replay*

CBC: Cipher Block Chaining

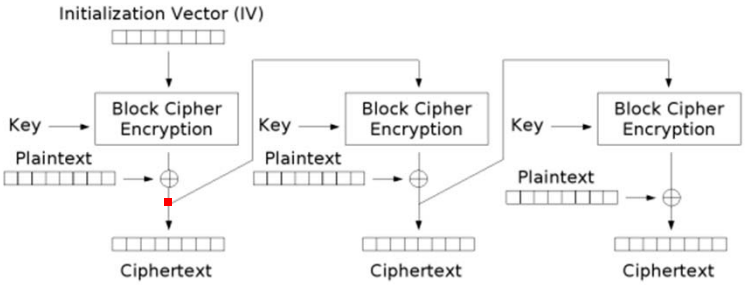
CBC: *Cipher Block Chaining*



- O texto em claro é "XOR" com o texto cifrado do bloco anterior antes de ser cifrado
- Reduz risco de replicação de padrões
- Initialization Vector (IV): usado no 1º bloco (necessário para decifrar)
- *Padding*: bits para compor blocos inteiros do tamanho requerido pelo algoritmo

CFB: Cipher Feedback

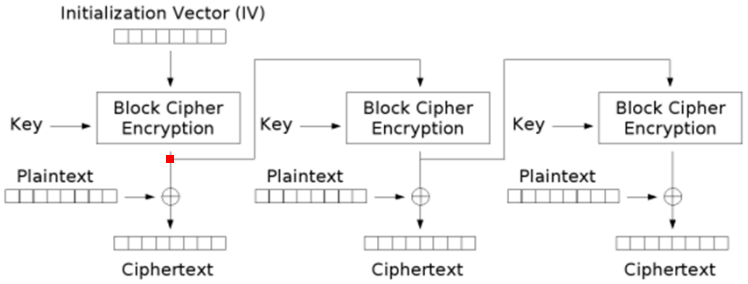
CFB: *Cipher Feedback*



- Transforma cifra por blocos em cifra contínua
- Vantagens sobre o CBC:
  - a cifra de bloco só é utilizada na direcção de cifrar (independentemente de a operação ser cifrar ou decifrar) o que simplifica a sua implementação
  - a mensagem não necessita de ser "padded" para um múltiplo do tamanho do bloco porque o algoritmo trabalha com qualquer quantidade de bytes

OFB: Output Feedback

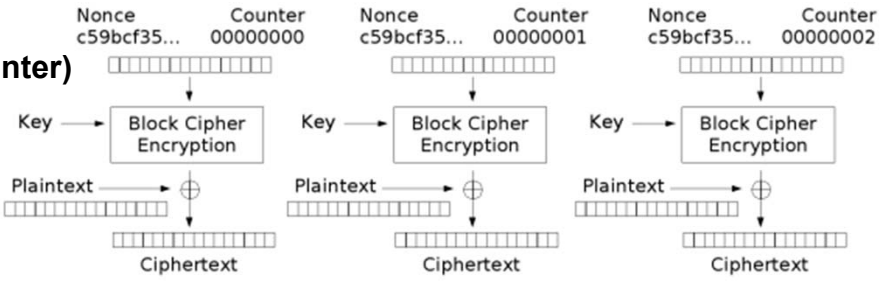
OFB: *Output Feedback*



- Mantêm as vantagens do CFB e ainda acrescenta outra
  - a mensagem não é utilizada no bloco seguinte, o que implica que as operações de cifra de bloco podem ser feitas antecipadamente permitindo que o XOR seja realizado em paralelo assim que o texto (mensagem ou mensagem cifrada) estiver disponível

CTR: Counter Mode

CTR (counter)



- Modo de cifra padrão para o AES
- Nonce+contador devem diferentes em cada operação de cifra
  - Caso seja usada a mesma chave e o mesmo nonce+contador para cifrar dois conteúdos diferentes, eles serão cifrados com duas chaves contínuas iguais ☹

Modos de cifra por blocos - comparação

- Reforço de segurança
  - Esconder padrões de texto original (exemplo: ECB ☹)
  - Confusão na entrada da cifra
    - exemplo : CBC com realimentação de bits do criptograma
  - Possibilidade de reutilização de uma chave de cifra
    - transformação em cifras contínuas; CFB, OFB, CTR ☹
  - Alteração determinística do texto em claro através da manipulação do criptograma
    - cifra contínua poderá ser fácil; pensar nos CFB, OFB, CTR ???
- Otimização
  - Efetuar pré-processamento
    - Exemplos: OFB e CTR
  - Paralelização do modo de cifra
    - Exemplos: ECB, CTR
- Tolerância a faltas
  - Propagação de erros
    - Exemplo: ECB – erro num bit apenas afeta o respetivo bloco
  - Recuperação de sincronismo de perda de bits

Modos de cifra por bloco - padding

- Modos de cifra: ECB e CBC
  - Iso10126
    - Exemplo: Blocos de 8 bytes e texto em claro 0x616263
    - Texto com padding 0x616263**????????05**
  - PKCS7
    - Exemplo: Blocos de 8 bytes e texto em claro 0x616263
    - Texto com padding 0x616263**0505050505**
  - Bit padding
    - 1011 1001 1101 0100 0010 011**1 0000 0000**

Reforço de segurança

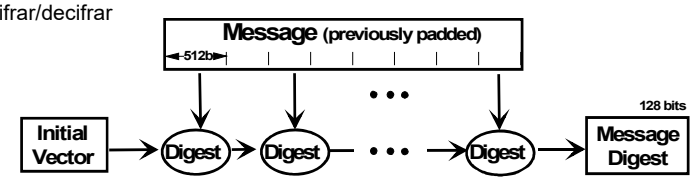
- Cifra múltipla
  - Cifra dupla
    - *Dual DES* usa duas chaves K_1 e K_2:
      - "chaves" *2n* bits
        - » Para cifrar: $C = E(K_2,E(K_1,P))$
        - » Para decifrar: $P = D(K_1,D(K_2,C))$
        - » Frágil: Pode ser quebrado usando 2^{n+1} cifrações (ao contrário do 2^{2n} esperados)
  - Cifra tripla
    - *Triple DES* (ou DES-EDE)
      - duas chaves K_1 e K_2:
        - » Para cifrar: $C = E(K_1,D(K_2,E(K_1,P)))$
        - » Para decifrar: $P = D(K_1,E(K_2,D(K_1,C)))$
        - » Também é mais frágil do que aparenta com relação a ataques de texto conhecido e texto escolhido
      - três chaves :
        - » Para cifrar: $C = E(K_3,E(K_2,E(K_1,P)))$
        - » Para decifrar: $P = D(K_1,D(K_2,D(K_3,C)))$
        - » Requer $O(2^{2n})$ cifrações e $O(2^n)$ de memória, com chaves de 56 bits, este ataque é inexequível

Sumário

- Introdução
- Criptografia e criptanálise
- Evolução da tecnologia de cifra
- Tipos de cifra
  - Cifras de transposição
  - Cifras de substituição
    - Monoalfabéticas
    - Polialfabéticas
- Conceitos teóricos
- Cifras contínuas
- Cifras modernas
  - Cifras simétricas
  - Cifras assimétricas
  - Cifras por blocos
    - Modos de cifra
    - Reforço de segurança
  - **Funções de síntese**
  - Autenticação
    - MACs
    - Assinaturas digitais

Síntese Segura ou *Digest* de Mensagens

- Objectivo
  - Produzem valores de dimensão constante (e pequena) a partir de entradas (mensagens, ficheiros, …) de dimensão variável
  - A função de compressão é aplicada de forma iterativa
    - 2 argumentos de entrada da função de compressão: síntese prévia, bloco a processar
  - Não servem para cifrar/decifrar



- Propriedades
  - Resistência à descoberta do texto original
    - Dada a síntese H, é muito difícil descobrir um texto M, tal que H = h(M)
  - Resistência à descoberta de um segundo texto original
    - Dado um Texto M, é muito difícil descobrir M'(M'≠M) tal que h(M)=H(M')
  - Resistência à colisão
    - É difícil descobrir dois textos quaisquer, M e M', M'≠M, tais que h(M) = h(M')
- Exemplos de Funções: MD5, SHA-1, SHA-256, SHA-3

Funções de síntese: Resistência à colisão

- Dimensão das sínteses
  - Com sínteses de 128 bits e mensagens 1000 bits: ~ 2^{872} mensagens dão origem à mesma síntese
  - Síntese >= 128 bits
- Ataque do Aniversário (*Birthday attack*):
  - Baseia-se no "paradoxo do aniversário" e é usado para encontrar um par de mensagens com a mesma síntese (colisão)
  - Para sínteses de n bits, o atacante deve tentar **aproximadamente $2^{n/2}$ mensagens**

Síntese Segura - MD5

- Proposto por Ron Rivest (investigador do MIT)
- O último de uma série de funções MD2, MD4, …
- Produz um valor de síntese de 128 bits
- Até recentemente era a função de síntese mais usada
  - recentemente foram encontradas falhas tanto através de ataques de força bruta quanto por criptanálise
  - especificado num padrão IETF (RFC1321)

- **Hoje em dia não é recomendada a sua utilização porque foi demonstrado que é possível encontrar colisões**

Síntese Segura- *Secure Hash Algorithm* (SHA)

- Várias variantes de algoritmos de hash propostos pelo NIST (e pela NSA) desde 1993
- Até recentemente, o SHA-1 era o mais utilizado, produzindo sínteses de 160 bits
- Baseia-se no desenho do MD4 com algumas diferenças (que aumentam muito a sua segurança)
- No mês passado foi descrito o primeiro ataque que demonstrava uma colisão (fev 2017), e por isso deve deixar de ser usada

- Existem diversas versões do SHA.
  - SHA-2: são baseadas nas mesmas ideias do SHA-1, mas são mais seguras na medida em que o tamanho da síntese produzida aumenta
  - SHA-3: utiliza um método diferente para a criação das sínteses, e resulta de um concurso para a criação de algoritmos de síntese organizado pelo NIST

Sumário dos algoritmos

Algorithm and variant		Output size (bits)	Max message size (bits)	Operations	Security (bits)
MD5		128	$2^{64} - 1$	And, Xor, Rot, Add, Or	<64 (collisions found)
SHA-0		160	$2^{64} - 1$	And, Xor, Rot, Add, Or	<80 (collisions found)
SHA-1		160	$2^{64} - 1$		<80 (collisions found)
SHA-2	*SHA-224* *SHA-256*	224 256	$2^{64} - 1$	And, Xor, Rot, Add, Or, Shr	112 128
	SHA-384 *SHA-512* *SHA-512/224* *SHA-512/256*	384 512 224 256	$2^{128} - 1$	And, Xor, Rot, Add, Or, Shr	192 256 112 128
SHA-3	*SHA3-224* *SHA3-256* *SHA3-384* *SHA3-512*	224 256 384 512	Unlimited	And, Xor, Rot, Not	112 128 192 256
	SHAKE128 *SHAKE256*	*d* (arbitrary) *d* (arbitrary)			min(*d*/2, 128) min(*d*/2, 256)

Sumário

- Introdução
- Criptografia e criptanálise
- Evolução da tecnologia de cifra
- Tipos de cifra
  - Cifras de transposição
  - Cifras de substituição
    - Monoalfabéticas
    - Polialfabéticas
- Conceitos teóricos
- Cifras contínuas
- Cifras modernas
  - Cifras simétricas
  - Cifras assimétricas
  - Cifras por blocos
    - Modos de cifra
    - Reforço de segurança
  - Funções de síntese
  - **Autenticação**
    - **MACs**
    - **Assinaturas digitais**

MAC - Message Authentication Code

- Funções de síntese – integridade
- Message Autentication Code – MAC
  - Usa chave simétrica partilhada
  - Como produzir MACs
    - Cifrar mensagem e síntese da mensagem
    - Cifrar síntese da mensagem
    - Fazer síntese da mensagem concatenada com a chave simétrica (HMAC)
    - Funções chaveadas (último criptograma gerado em modo CBC)
  - O que garante ?

HMAC - *Hash Message Authentication Code*

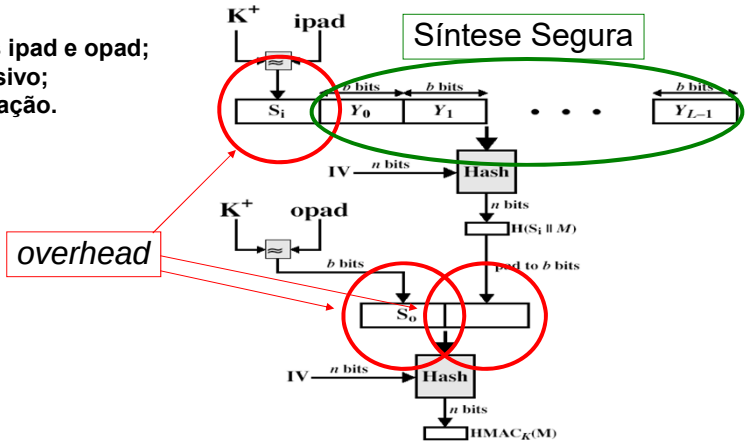
- Definido no RFC2104 do IETF
- Utiliza funções de síntese na mensagem
- Qualquer função de síntese segura pode ser usada no HMAC, como o MD5 (HMAC-MD5), SHA-1 (HMAC-SHA-1), etc.
- A segurança e a eficiência do algoritmo dependem da função de síntese segura usada
- A função de síntese pode ser substituída com o fim de melhorar segurança e/ou eficiência

HMAC - *Hash Message Authentication Code*

$H_k(m) = hash(k' \oplus opad \,||\, hash(k' \oplus ipad \,||\, m))$

onde:

- **Constantes ipad e opad;**
- **⊕ ou exclusivo;**
- **|| concatenação.**



Assinatura?

- MAC:
  - Alice e Beto compartilham uma chave *k*
  - Alice envia *m* || $H_k(m)$ para Beto
- Isto é uma assinatura digital?
  - Beto verifica que recebeu *m* de Alice
- Certo?

ERRADO!

- Isto não é uma assinatura digital!
  - Um terceira parte não pode determinar se foi Alice ou Beto quem gerou a mensagem
  - i.e., não satisfaz a propriedade de não-repudiação

Assinaturas Digitais

- Autenticidade
  - quem assinou é identificável univocamente pela assinatura
- Integridade
  - uma assinatura correcta num documento garante que este não é alterável sem detecção
- Não-reutilização
  - a assinatura ou parte do documento não é reutilizável em outro documento
- Não-repudiação
  - o assinante não pode negar a sua assinatura
- Não-forjamento
  - quem assinou é o próprio e fê-lo deliberadamente

Assinaturas Digitais

- Modo de operação
  - Cifrar com chave privada a síntese do texto original
  - Algoritmos de assinatura mais usados
    - RSA
    - DSA
- RSA
  - Dada a mensagem M: Assinatura = E(Kr, Hash(M))
  - Para maiores garantias posso concatenar outra informação ao M, como o identificador de quem assina e a data
- DSS/DSA
  - Proposta de solução específica para assinaturas digitais do NIST (1991)
  - Digital Signature Standard (DSS) e Digital Signature Algorithm (DSA)
    - derivado do algoritmo de assinatura ElGamal
    - usa a função de síntese SHA-1

Criptografia Híbrida

- Vamos juntar a criptografia simétrica com a assimétrica
- Cifra híbrida - cifrar
  - Gerar chave secreta aleatória
  - Cifrar texto em claro com chave secreta
  - Cifrar chave secreta com chave pública do destinatário
  - Garantias / Vantagens ?
- Cifrar e assinar
  - Assina texto em claro
  - Cifra texto em claro + assinatura (como no exemplo anterior)
  - Garantias / Vantagens ?
- Ambos garantem a autenticidade e/ou integridade?

Bibliografia

- Stallings & Brown, Computer Security: Principles and Practice, Third Edition, 2015
  - Leitura obrigatória: Capítulo 2
  - Leitura opcional: Capítulo 20 e 21

