Criptografia

Conceitos de base

Criptografia

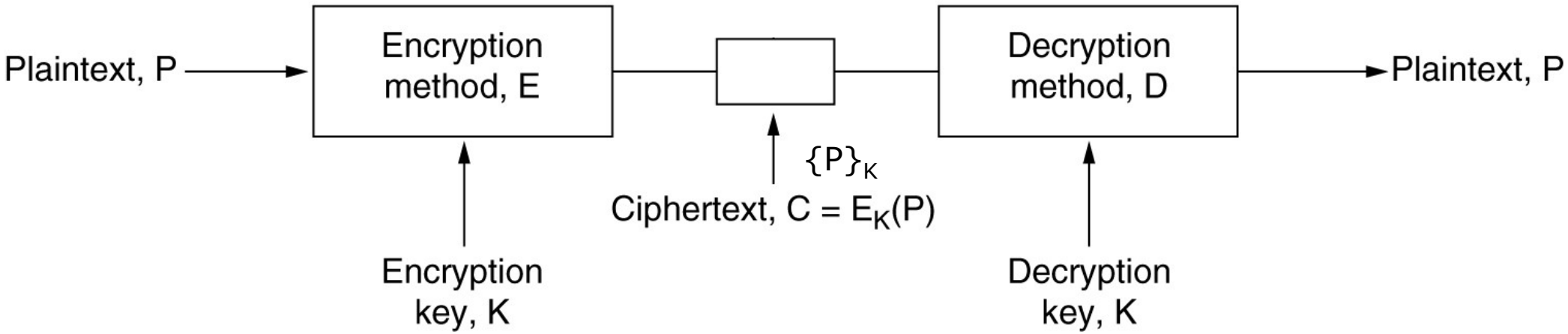
- A base da criptografia é conseguir que um grupo de pessoas transmita informação entre elas que ***seja ininteligível para todas as outras***
- Uma solução: ter um dialeto próprio
  - **Não é escalável, nem seguro**.
- Melhor solução:
  - Algoritmo que cifra a informação conhecido e uma chave que parametriza o algoritmo
  - **Algoritmo público, chave é segredo**
  - Análogo às fechaduras físicas...

Criptografia
Conceitos

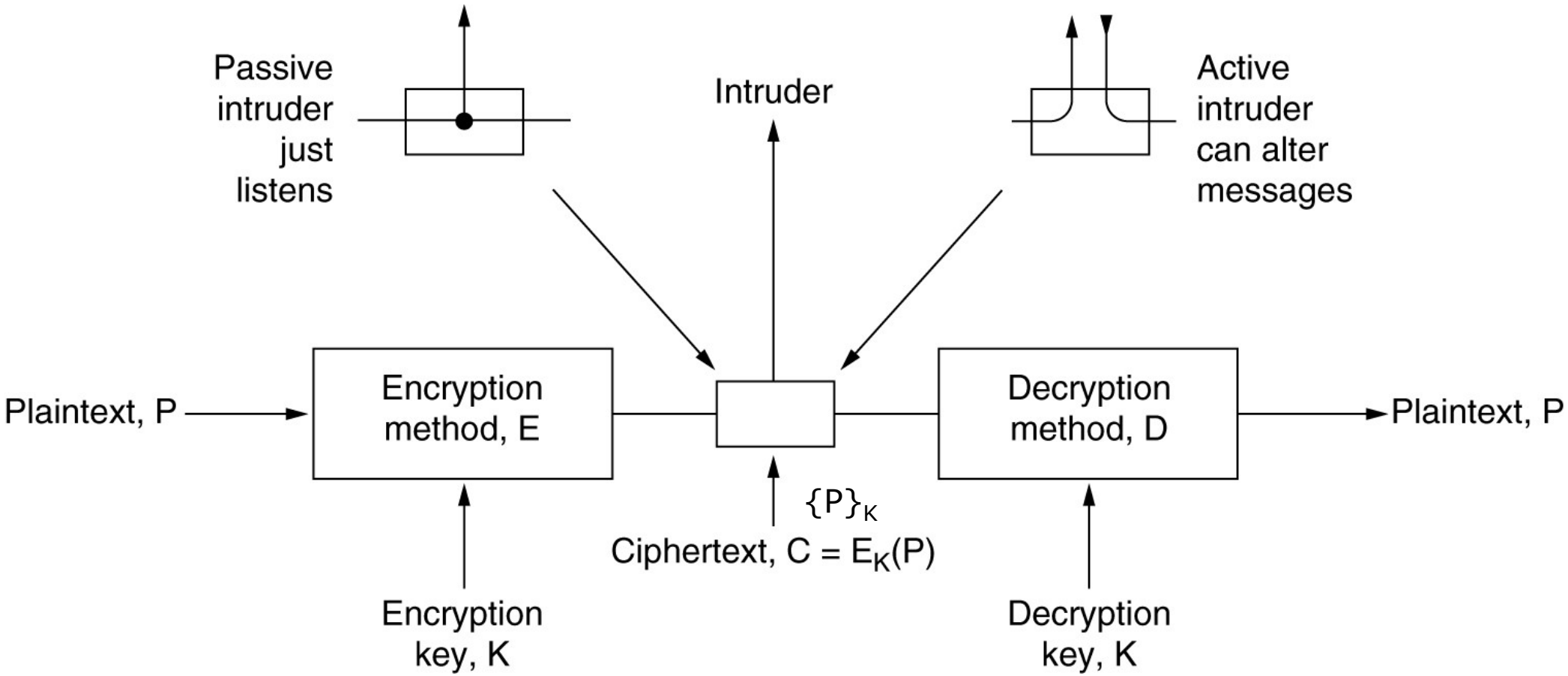
- Algoritmo que cifra
  - Função injectivas
  - Parametrizadas por uma chave
- Algoritmo que decifra
  - As cifras são reversíveis apenas por quem possuir o algoritmo inverso
  - Parametrizado por chave inversa
- Nomenclatura:

$M \rightarrow \{M\}_K$: cifra da mensagem M com a chave K, gerando um criptograma

Modelo de Comunicação Cifrada



Modelo de Comunicação Cifrada com Intrusos



Métodos genéricos de ataque a funções de cifra

- Dependem de em que situação o atacante está
  a) Só tem acesso a mensagens cifradas
  b) Tem acesso a amostras de um texto em claro e cifrado
  c) A partir de qualquer texto original, pode gerar o cifrado

Em qual se encontra cifra assimétrica?

- Em b) e c), é sempre possível o ataque exaustivo (*brute-force*)
  - Atacante itera todas as chaves possíveis até que cifra do texto original resulte no cifrado

Como prevenir?

- Em c), é também possível o *chosen-plaintext attack (CPA)* caso a mensagem cifrada seja pequena
  - Quando mensagem cifrada C é pequena, itera-se todas as mensagens M até se obter C

Como prevenir?

Criptografia: Ataque de força bruta (*brute force*)

- Admitindo que o algoritmo não permite ataques simples, obter a informação através de teste sistemático das chaves

- A dimensão da chave é decisiva
  - Em média, é preciso percorrer metade do espaço de procura
  - Uma chave de pequena dimensão pode ser facilmente encontrada por teste sistemático

Criptografia com Segurança Total vs Prática

- As funções de cifra são consideradas **totalmente** seguras se:
  - Independentemente do tempo e do poder computacional envolvido, a chave não puder ser descoberta

- Normalmente são **praticamente** seguras
  - O valor da informação não justifica o investimento computacional (em máquinas especiais)
  - Temporalmente limitada a sua validade e muito inferior ao tempo necessário para decifrá-la com a tecnologia existente

Criptografia: simétricas vs assimétricas

- Cifras simétricas
  - Normalmente usam técnicas de substituição e difusão
  - São normalmente muito mais rápidas que as assimétricas
- Cifras assimétricas
  - Normalmente usam operações matemáticas
  - A sua segurança baseia-se na complexidade de certas operações matemáticas
    - Logaritmo modular
      - $Y = a^X \bmod b$; Dados a, b e Y, calcular X
    - Factorização de grandes números
      - Y = ab, a e b primos; Dado Y, calcular a ou b

Criptografia Simétrica

Cifras históricas

- Cifra de César
  - Shift by 3
- Cifra de substituição
  - A -> C
  - …
- Cifra de Vigener (1500s)
  - + mod 26
- Máquina de rotores (1870-1943)
  - Rotor único: Hebern
  - 3-5 rotors: Enigma

Cifra simétrica

- Substituição
  - Mono-alfabética
  - Poli-alfabética
- Exemplo Mono-alfabético
  - Chave – troia

```
ABCDEFGHIJLMNOPQRSTUVXZ
TROIABCDEFGHJLMNPQSUVXZ
```

  - Problema?

Cifra Simétrica

- Poli-alfabético
  - Procura que as distribuições sejam combinadas de forma a que não existam caracteres que sejam mais frequentes
  - Difundir no criptograma a mensagem
- Exemplo: Tabelas de Vigenère

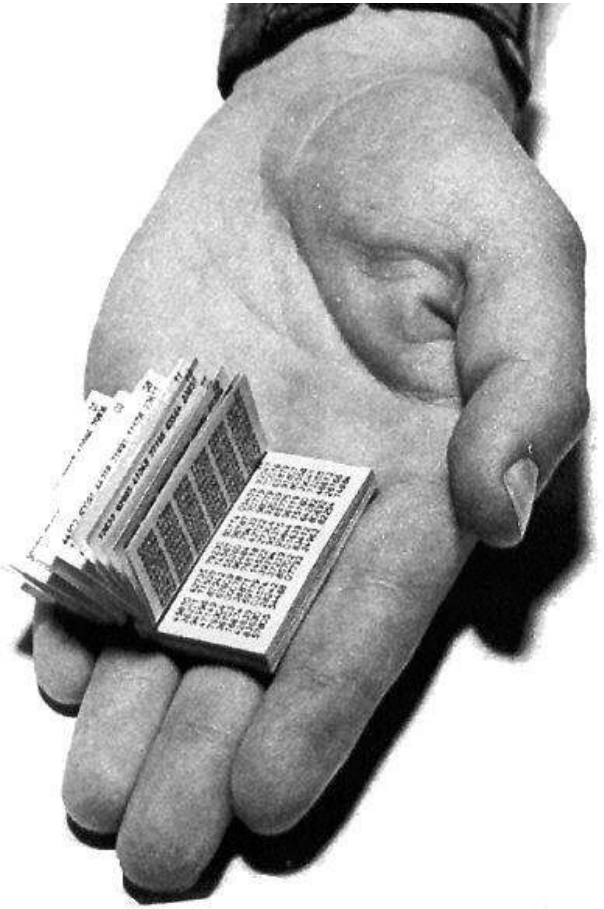


Table 2-5 Vigenère Tableau

	0					5					10					15					20					25	
	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*	π
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	0
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	1
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	2
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	3
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	4
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	5
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	6
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	7
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	8
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	9
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	10
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	11
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	12
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	13
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	14
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	15
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	16
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	17
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	18
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	19
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	20
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	21
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	22
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	23
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	24
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	25

Exemplo de Cifra com a Tabela de Vigenère

- Vamos, supor que se pretende cifrar uma mensagem em claro (*plaintext*) :
  - ATTACKATDAWN
- O cifrador escolhe a chave e repete-a até que tenha o tamanho da mensagem
  - Vamos usar "LEMON": LEMONLEMONLE
- A primeira letra da mensagem, A, é cifrada usando o alfabeto na linha L, que é a primeira letra da chave. Na tabela de Vigenère corresponde à linha L e à coluna A.
  - Da mesma forma para a segunda letra da mensagem: a linha E e a coluna T resulta X.
  - A restante mensagem é cifrada da mesma forma
- Mensagem:
  - ATTACKATDAWN
- Chave:
  - LEMONLEMONLE
- Criptograma
  - LXFOPVEFRNHR

Cifra simétrica

- Objetivo
  - Confundir – operações não destrutivas que permitam alterar o significado da mensagem em aberto misturando-o com a chave
  - Difundir – fazer com que as alterações se difundam a toda a mensagem cifrada para não ser alvo de análise estatística de padrões

- Operações usuais:
  - XOR (eXclusive-OR)
  - *Shift* (deslocamento)
  - Permutação de bits

Cifra simétrica contínua vs blocos

- Cifras contínuas (*Stream ciphers*)
  - Cifram dados que chegam “sem parar”
  - Existe uma *key stream*

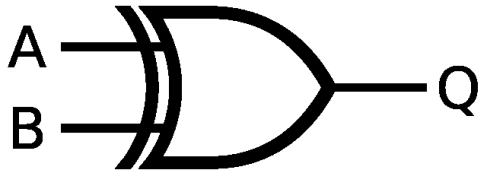
- Cifras em blocos (*Block ciphers*)
  - Cifram um bloco de cada vez
  - Podem ser encadeadas para esconder padrões nos criptogramas

Cifra Simétrica Contínua

XOR (eXclusive OR)

Addition Modulo 2

p	q	$p + q$
0	0	0
0	1	1
1	0	1
1	1	0

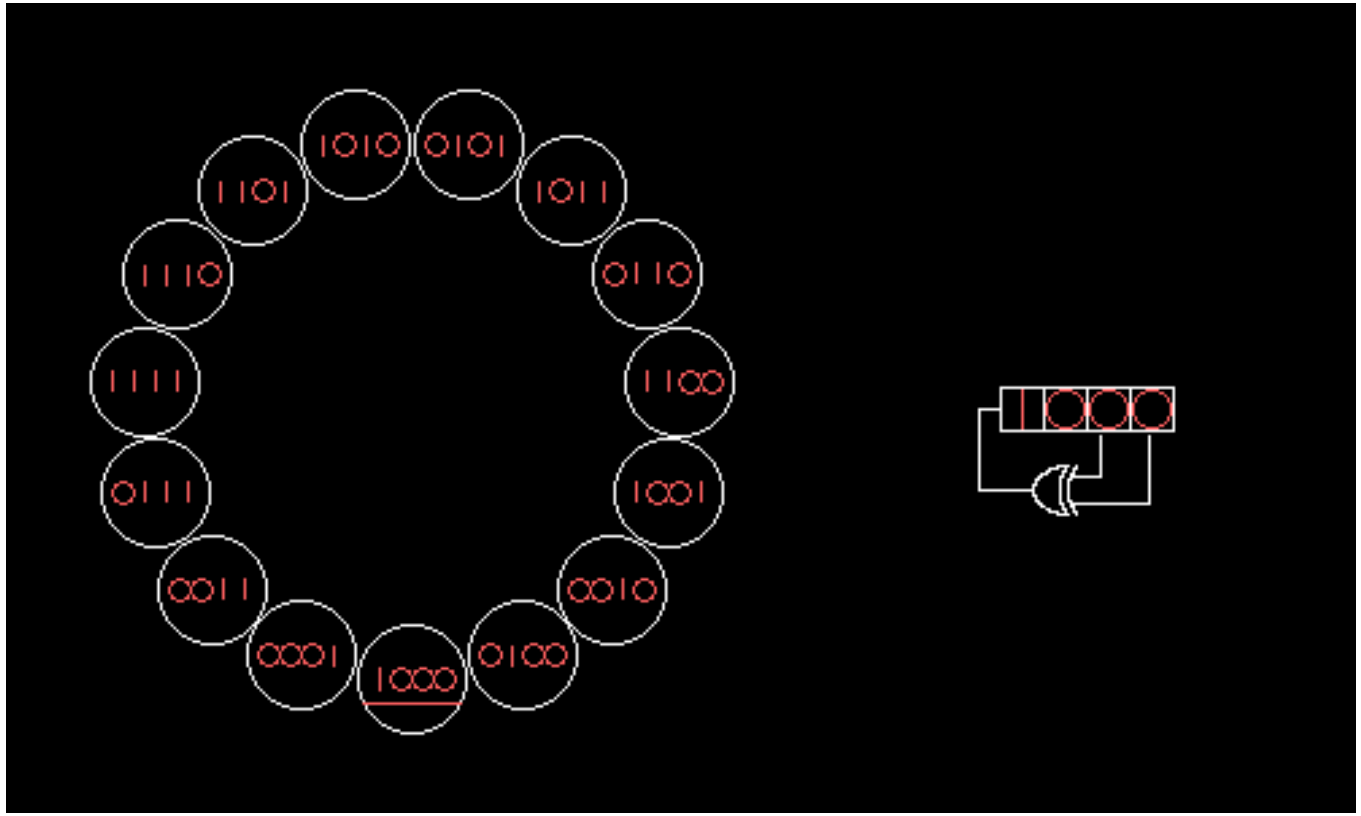


One-time pads

- Chave de grande dimensão não repetida
  - Substituição poli-alfabética
- O emissor usa a parte da chave que necessita para cifrar a mensagem
- O recetor usa a mesma parte da chave estando ambos sincronizados sobre que parte já utilizaram
- Totalmente seguro, mas... como distribuir a chave?
  - Uma aproximação a *one-time pads* nos computadores são geradores de números aleatórios
  - Que funcionam a partir de chave (limitada) distribuída inicialmente

Linear Feedback Shift Register (LFSR)

- Gerador de números pseudo-aleatórios



Bits 2 e 3

Cifra Simétrica por Blocos

Data Encription Standard - DES

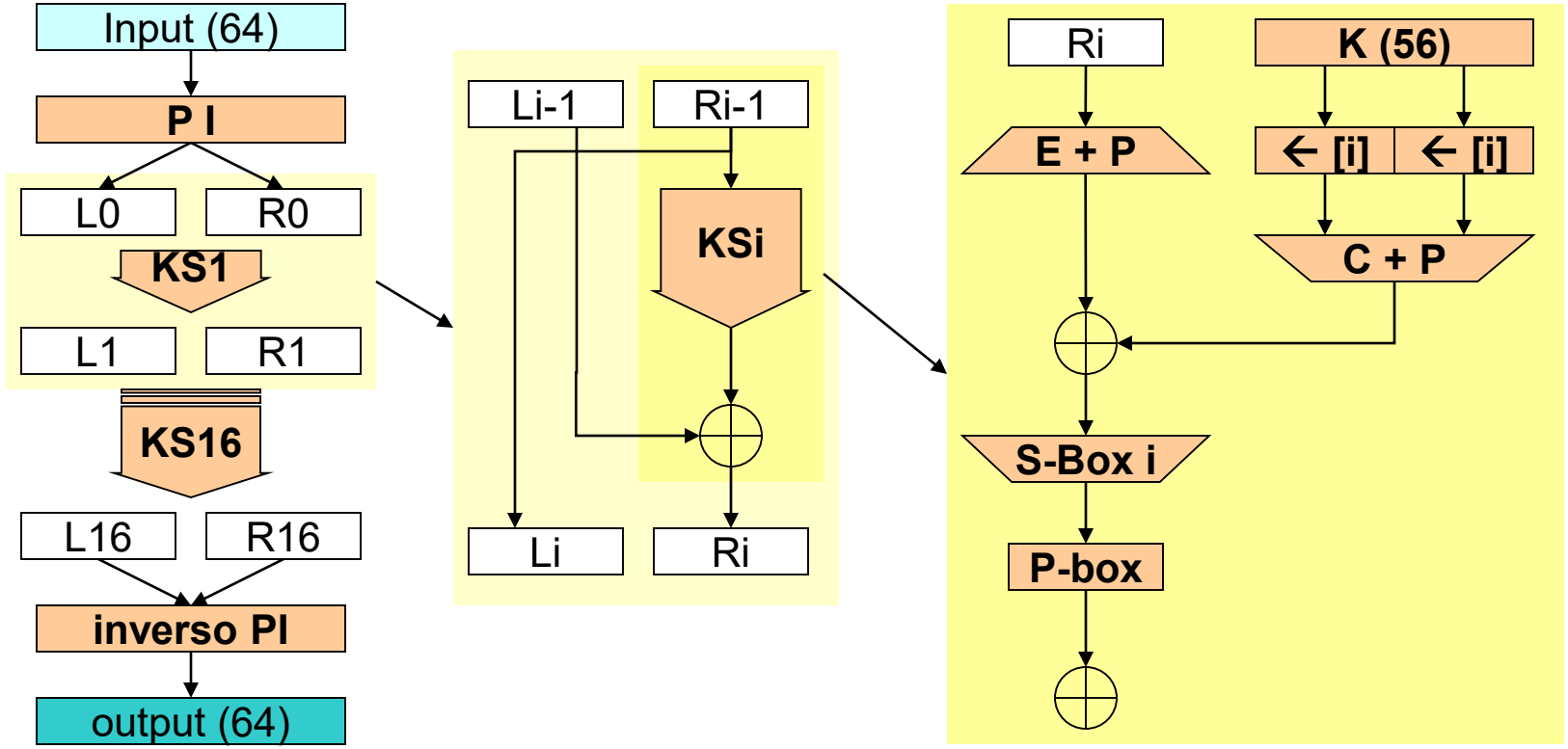
- 1970 - O National Bureau of Standards (NBS) dos EUA reconheceu a necessidade de um algoritmo padrão para cifra na sociedade civil
- 1972 – O NBS abriu um concurso para uma novo algoritmo que devia ter várias características, entre elas:
  - Alto nível de segurança
  - Completamente especificado e fácil de perceber
  - O algoritmo devia ser público, a sua segurança não vinha de ser secreto
  - Adaptável a diversas utilizações
  - Fácil de realizar em dispositivos electrónico
- 1974 - Os primeiros resultados foram desencorajadores e houve um segundo concurso
- Desta vez foi considerada aceitável a proposta do algoritmo de cifra Lucifer desenvolvido pela IBM
- 1976 – Depois de análise pelo DoD em particular pela NSA foi aceite como standard nos EUA

Data Encription Standard - DES

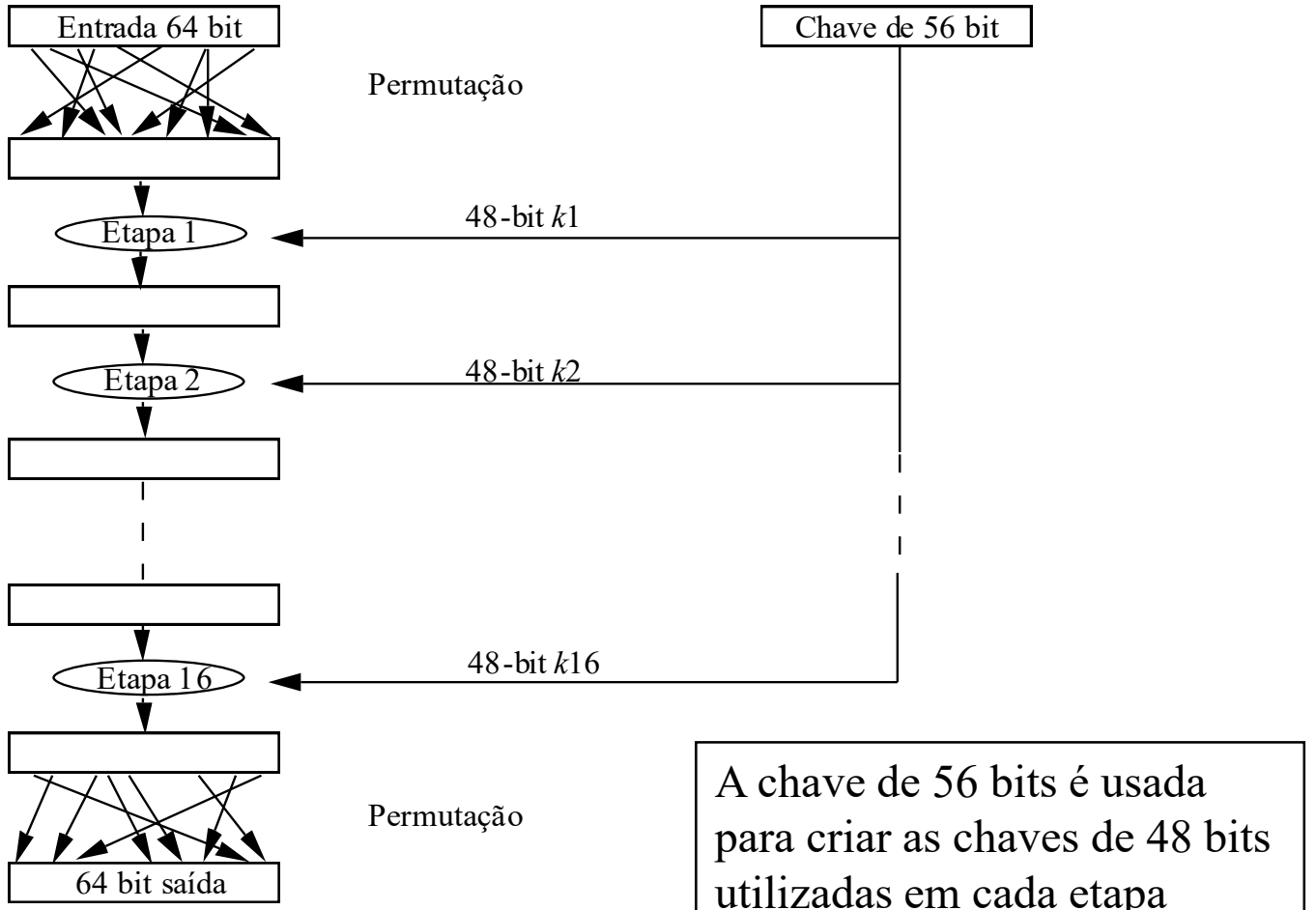
- Blocos de 64 bits
- Aplica funções de permutação e substituição a cada bloco
- 16 etapas e duas permutações totais
- Chave de 56 bits, desdobrada em chaves de 48 bits para cada etapa
- Pode ser realizado em *software* ou em *hardware*

DES

- Substituição, Permutação, Compressão e Expansão

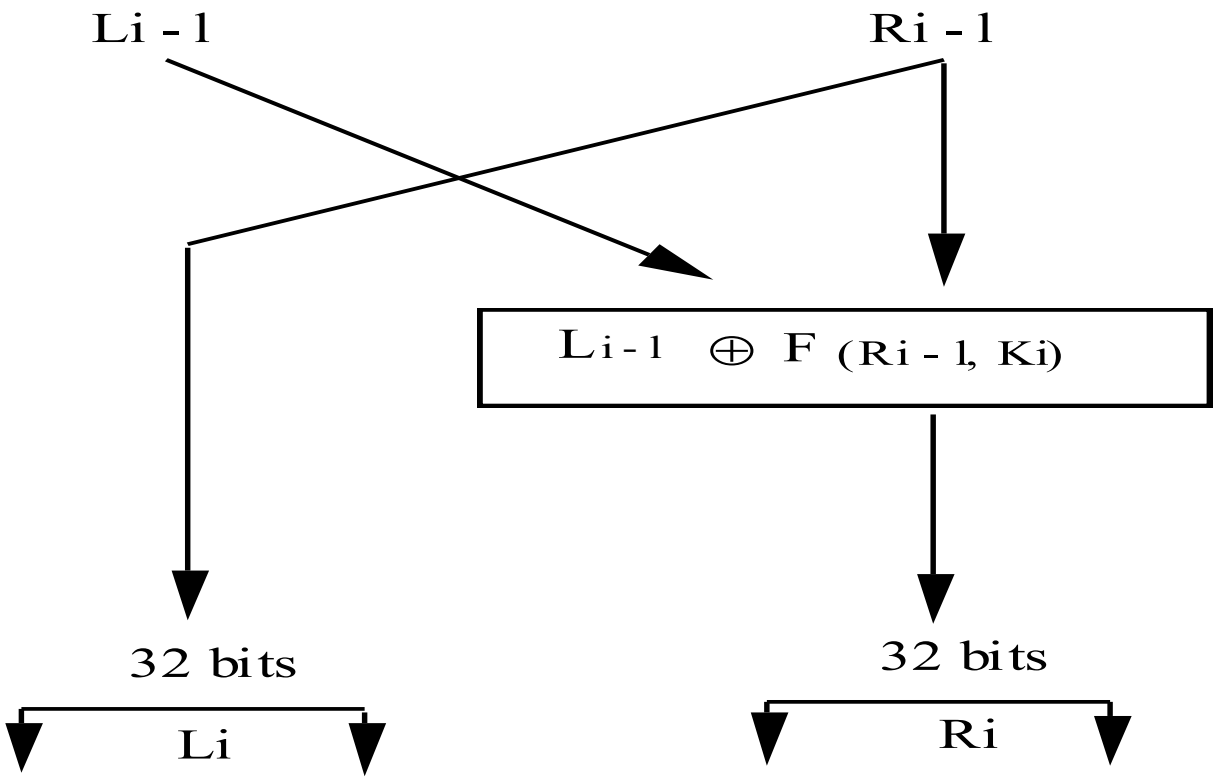


Algoritmo do DES



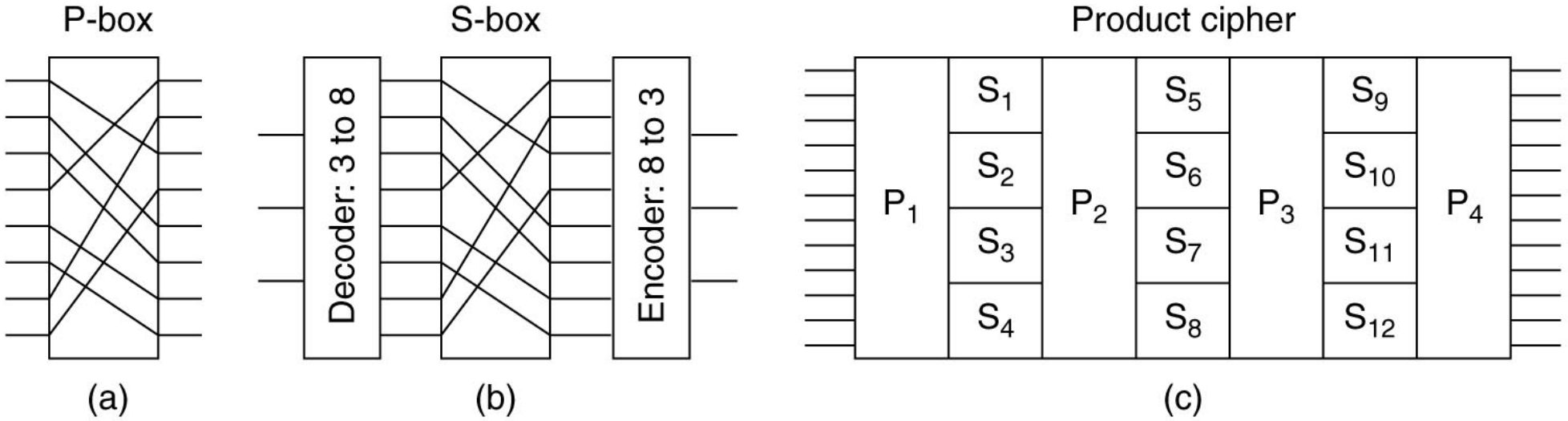
A chave de 56 bits é usada para criar as chaves de 48 bits utilizadas em cada etapa

Função de base do DES



Técnicas Elementares de Criptografia Simétrica

- Substituição - dificultar a descoberta da forma como a mensagem e a chave foram utilizadas na transformação da informação.
- Permutação - difundir a informação uniformemente pelo texto cifrado.



Exemplo de uma S-box

- Os 48 bits de cada etapa são transformados por 8 *substitution boxes* – S-Boxes
- Podem ser vistas como uma função com 6 bits de entrada e 4 de saída
- A representação interna da função é na forma de uma tabela que a partir de 4 bits de entrada escolhe com base em dois bits (4+2) um valor de saída de 4 bits

S_5		Middle 4 bits of input															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1100**	**1011**	**1100**	**1101**	**1110**	**1111**
Outer bits	**00**	0010	1100	0100	0001	0111	1100	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1100	0011	1001	1000	0110
	10	0100	0010	0001	1011	1100	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1100	0100	0101	0011

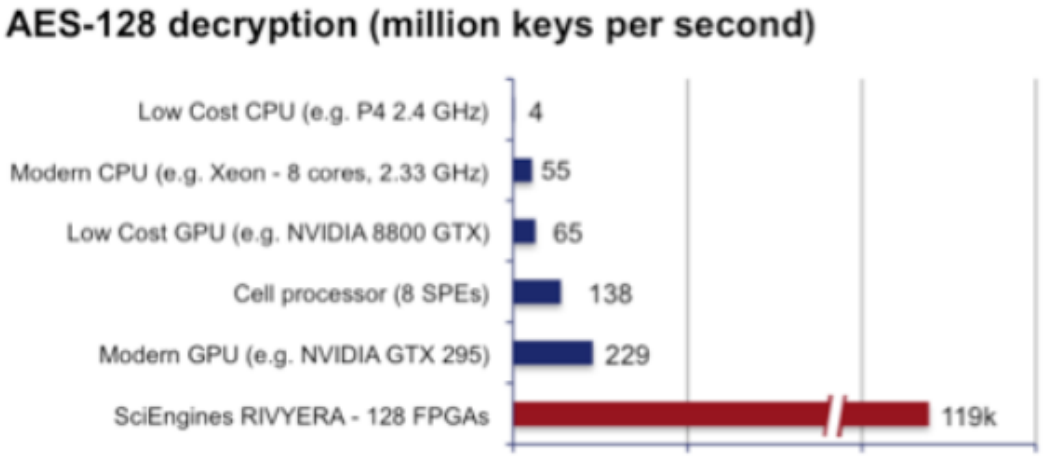
Chave do DES

- Só há registos de quebra por teste sistemático (*brute-force*) da chave
- Desde a sua publicação que a chave de 56 bits é considerada **insuficiente**, permitindo que o sistema seja alvo de ataques sistemáticos.
- Com o rápido aumento do desempenho das máquinas, esta questão torna-se cada vez mais preocupante.
- [Kaufman95] considera que as chaves deveriam crescer
  - 1 bit a cada dois anos
- O algoritmo inicial da IBM tinha chaves de 112
  - Especula-se que a NSA esteve por trás da sua redução

Chave do DES

- Em 2006 um computador dedicado designado de COPACOBANA construído por $10,000 quebrou o DES com ataques de força bruta em 8,7 dias
- Em 2009 conseguia-se o mesmo em apenas 6 dias.

COPACOBANA and its successor **RIVYERA** offer a massively parallel platform allowing more parallel „primitive operations“ per second than any other platform in the market. An example of the new RIVYERA's performance for e.g. encoding 128 bit AES compared to other commonly used hardware is illustrated below. In case you want to assess different encryption algorithms' performance for finding the one with the most suitable security profile for you: Please keep in mind that AES has a too large key space for exhaustive search attacks against correctly implemented encryptions with computer-generated keys - no matter if cutting-edge FPGA hardware is used. But, human-generated passwords are still at risk, if the attacker has sufficient processing resources available and below figures may provide an approximate guide for selecting a minimum length and complexity of passwords.



Funções de Substituição

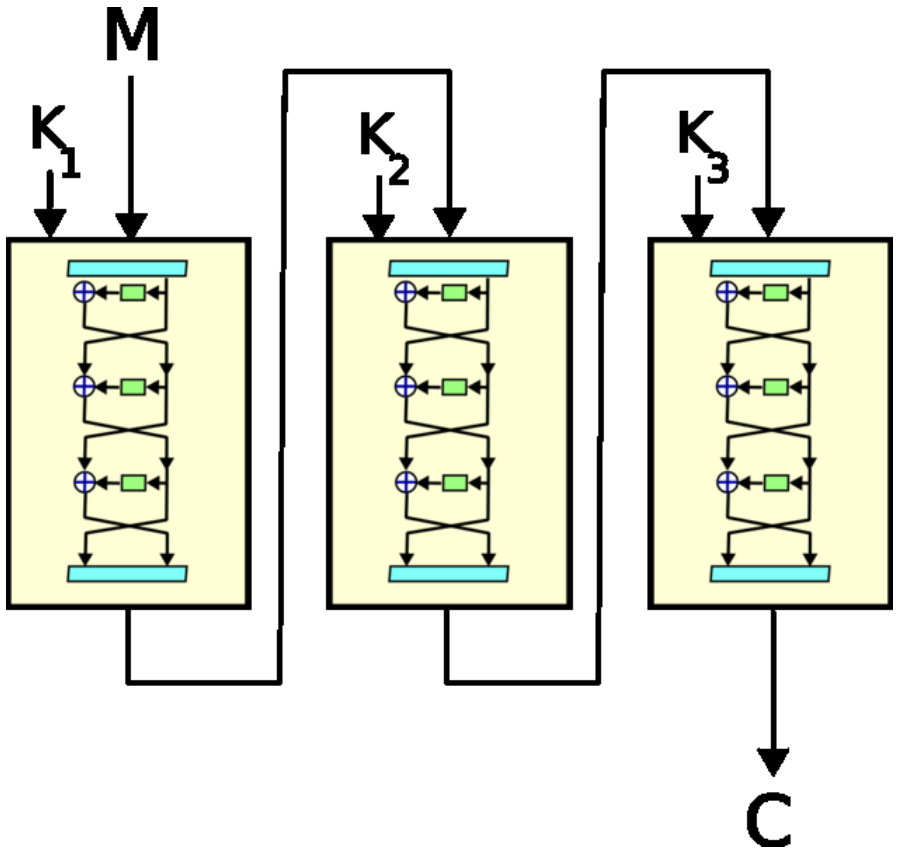
- A chave e a palavra de 32 bits são entradas de uma função que mistura os respetivos bits produzindo uma palavra de 32 bits.
- Cada etapa pode ser decomposta em duas operações,
  - A parte mais significativa é uma cópia dos 32 bits menos significativos da entrada (Li = Ri-1)
  - Na outra metade efetua-se um “ou-exclusivo” dos 32 bits mais significativos, com o resultado da função F que tem por entradas os 32 bits menos significativos e parte da chave Ki
- Grande parte da complexidade do algoritmo reside nas funções de substituição
- Os detalhes das funções e das diferentes etapas são conhecidos

Algoritmos de Cifra Simétrica

- DES
- Triple DES
- RC4
- RC5
- IDEA
- Blowfish
- AES – *Advanced Encription Standard* – norma dos EUA, com chaves de 128, 196 e 256 bits

DES Triplo

- Resolução para o problema da dimensão da chave
-Com 3 chaves de 56 bits diferentes, DES triplo consegue segurança efectiva de 112 bits (< 168 bits)



Algoritmos de Cifra Simétrica: Comparativo

Cipher	Author	Key length	Comments
Blowfish	Bruce Schneier	1–448 bits	Old and slow
DES	IBM	56 bits	Too weak to use now
IDEA	Massey and Xuejia	128 bits	Good, but patented
RC4	Ronald Rivest	1–2048 bits	Caution: some keys are weak
RC5	Ronald Rivest	128–256 bits	Good, but patented
Rijndael	Daemen and Rijmen	128–256 bits	Best choice
Serpent	Anderson, Biham, Knudsen	128–256 bits	Very strong
Triple DES	IBM	168 bits	Second best choice
Twofish	Bruce Schneier	128–256 bits	Very strong; widely used

AES

- Rijndael - **Advanced Encryption Standard** (**AES**)
- Fonte: Computer Networks, Andrew Tanenbaum

Exemplo académico de cifra simétrica: TEA

- Algoritmo muito simples
- Razoavelmente rápido
- Descrito no livro da cadeira
  - Coulouris

Exemplo de cifra simétrica: TEA

```
void encrypt(unsigned long k[], unsigned long text[]) {
    unsigned long y = text[0], z = text[1];                          1
    unsigned long delta = 0x9e3779b9, sum = 0; int n;                2
    for (n= 0; n < 32; n++) {                                        3
        sum += delta;                                                4
        y += ((z << 4) + k[0]) ^ (z+sum) ^ ((z >> 5) + k[1]);        5
        z += ((y << 4) + k[2]) ^ (y+sum) ^ ((y >> 5) + k[3]);        6
    }
    text[0] = y;  text[1] = z;                                       7
}
```

32 etapas.
Técnicas base:
shift de bits, XOR, soma, dependentes da chave k

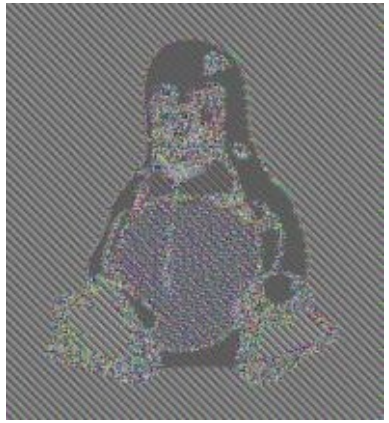
Exemplo de cifra simétrica: TEA

```
void decrypt(unsigned long k[], unsigned long text[]) {
    unsigned long y = text[0], z = text[1];
    unsigned long delta = 0x9e3779b9, sum = delta << 5;  int n;
    for (n= 0; n < 32; n++) {
        z -= ((y << 4) + k[2]) ^ (y + sum) ^ ((y >> 5) + k[3]);
        y -= ((z << 4) + k[0]) ^ (z + sum) ^ ((z >> 5) + k[1]);
        sum -= delta;
    }
    text[0] = y; text[1] = z;
}
```

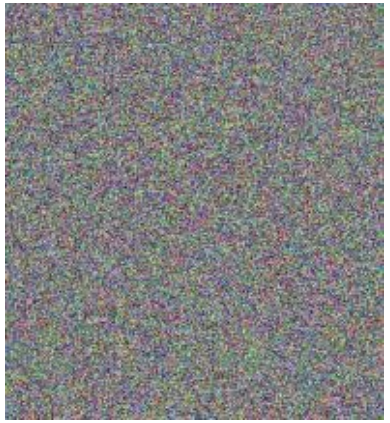
Cifra Simétrica por Blocos com Realimentação

Por Blocos versus Contínuas: Exemplo

Original



Cifra Por Bloco



Cifra Contínua

Fonte: Wikipedia

Modos de cifra

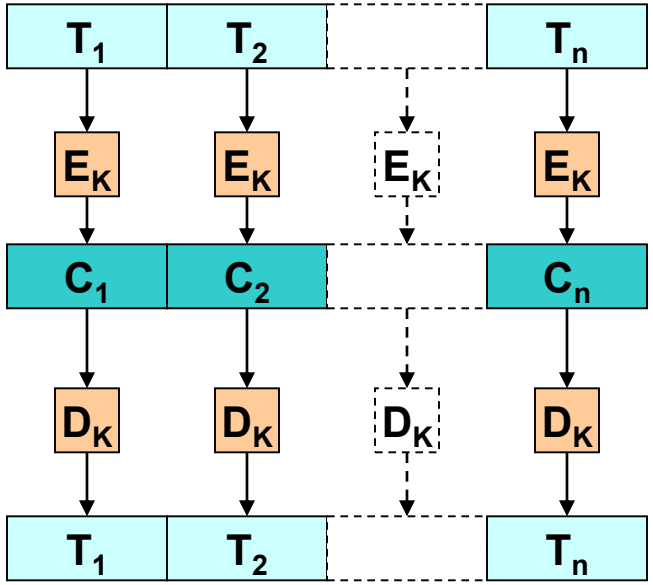
- Inicialmente apresentados para o DES
  - ECB (Electronic Code Book)
  - CBC (Cipher Block Chaining)
  - Stream Cipher
- Podem ser usados por outras cifras por blocos

Modos de cifra: ECB vs CBC

Electronic Code Book

$C_i = E_K(T_i)$

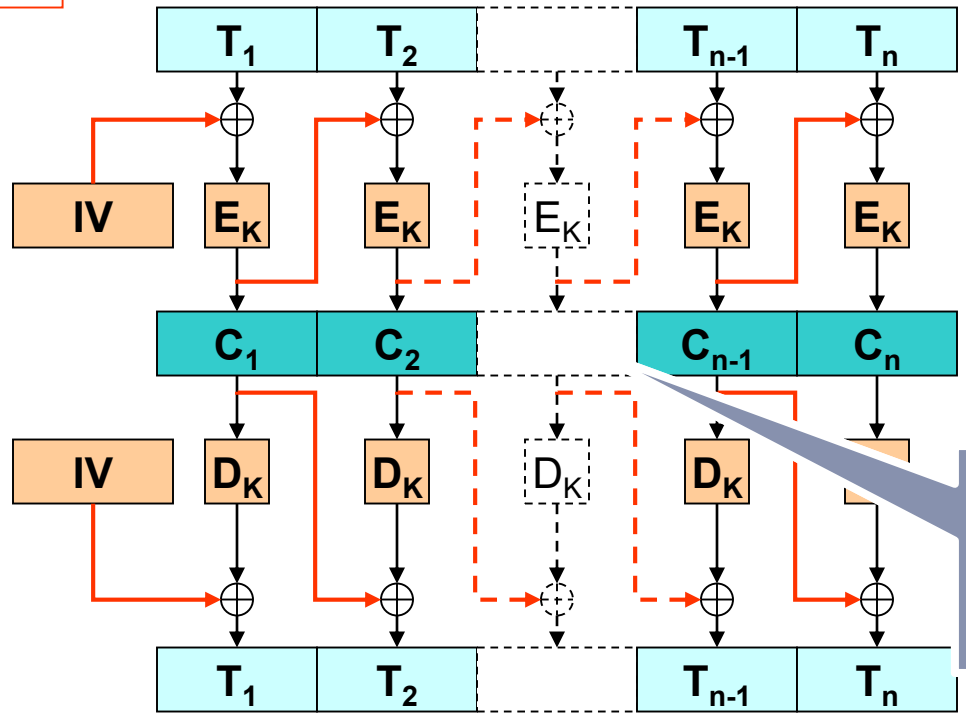
$T_i = D_K(C_i)$



Cipher Block Chaining

$C_i = E_K(T_i \oplus C_{i-1})$

$T_i = D_K(C_i) \oplus C_{i-1}$



Necessidade de um IV - Initialization Vector para o primeiro bloco

Se C_i se perde na rede, consegue decifrar C_{i+1}?

Modos de cifra: OFB e CFB

Output Feedback (autokey)

$C_i = T_i \oplus E_K(S_i)$

$T_i = C_i \oplus E_K(S_i)$

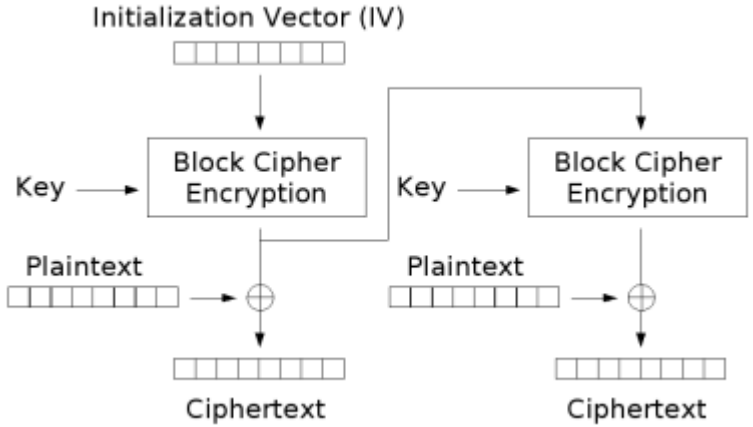
$S_i = f(S_{i-1}, E_K(S_{i-1}))$

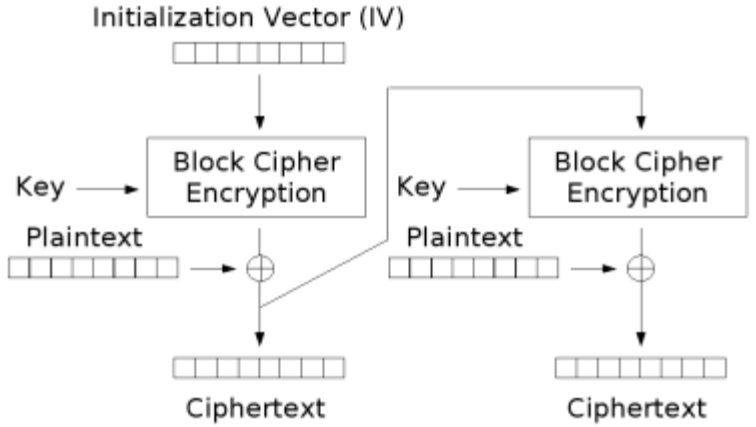
Ciphertext Feedback

$C_i = T_i \oplus E_K(S_i)$

$T_i = C_i \oplus E_K(S_i)$

$S_i = f(S_{i-1}, C_i)$





Criptografia Assimétrica

Aritmética Modular da Multiplicação

X*Y mod N

N=10

X \ Y	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9
2	0	2	4	6	8	0	2	4	6	8
3	0	3	6	9	2	5	8	1	4	7
4	0	4	8	2	6	0	4	8	2	6
5	0	5	0	5	0	5	0	5	0	5
6	0	6	2	8	4	0	6	2	8	4
7	0	7	4	1	8	5	2	9	6	3
8	0	8	6	4	2	0	8	6	4	2
9	0	9	8	7	6	5	4	3	2	1

Funções injectivas

O inverso multiplicativo é a solução da equação:

$Y*Y^{-1} \bmod N = 1$

Exemplo:
$Y=3$, $Y^{-1}=7$

Para cifrar (Valor 9):
C = 9*3 mod 10 = 7

Para decifrar:
T = 7*7 mod 10 = 9

Algoritmos de cifra assimétrica

- Diffie Hellman
- RSA
- DSS – baseado ElGamal
- Curvas elípticas

RSA - Rivest Shamir Adleman

- Algoritmo de cifra de chave pública mais divulgado
- Patente expirou
- Enquanto era válida, os autores permitiram aos *browsers* utilizar o algoritmo sem pagar
  - Desde que reconhecessem a sua empresa (VeriSign) como autoridade para gerar certificados

RSA - Rivest Shamir Adleman

- O método baseia-se na existência de duas chaves
- Qualquer uma pode ser usada para cifrar ou decifrar
- Uma das chaves é **pública** Kp (todos conhecem) e outra é **privada** ou secreta Ks
- O emissor cifra a mensagem efectuando a exponenciação da informação elevada à chave Kp e efectuando o módulo com N:

$$\{M\}_{Kp} = M^{Kp} \bmod N$$

- O receptor decifra a informação efectuando a exponenciação com a outra chave e calculando o módulo N:

$$M = (\{M\}_{Kp})^{Ks} \bmod N$$

Fundamento do RSA

- P,Q números primos da ordem de 10^{100}
- N = P*Q
- Z = (P-1)*(Q-1)
- Kp e Ks são coprimos com Z tais que Kp*Ks = 1 mod Z

Exemplo do cálculo das Chaves

1- Escolhem-se dois números primos P e Q e calcula-se N e Z,

- Vamos supor P = 13, Q = 17:
- **N = P * Q = 13 x 17 = 221**
- **Z = (P - 1)*(Q - 1) = 12 x 16 = 192**

2 - A chave K_p é um número co-primo com Z.

Neste caso, Z = 2*2*2*2*2*2*3, pelo que podemos escolher $K_p = 5$

3 - Para calcular K_s é necessário resolver a equação $K_p * K_s = 1 \bmod Z$,

- **K_s *5= 1 mod 192**
- **K_s * 5 = 1, 193, 385, …**
- **K_s = 385:5 = 77**

Chaves

- São trocados N e Kp que constituem a chave pública
- N e Ks são a chave privada

Cifra/Decifra em RSA

- Cifra por blocos de dimensão k, em que $2^k < N$
  - No nosso exemplo, k=7
- Para cifrar mensagem em claro M:

$$\{M\}_{Kp} = M^{Kp} \bmod N$$

- Para decifrar mensagem cifrada C:

$$\{C\}_{Ks} = C^{Ks} \bmod N$$

Quebrar a chave privada sabendo a chave pública?

- Se atacante sabe Kp e N,
  como consegue descobrir a chave privada?
  - Para saber Ks é preciso saber Z
    - (ver slides de geração de chaves)
  - Para saber Z é preciso saber os dois números primos P e Q tal que PxQ=N
- Se N > 10^{200}, em 1978, Rivest considerava que para computadores que executassem 1 MIPS levariam 4 mil milhões de anos
- Mas houve contínua evolução dos computadores e dos métodos de cálculo…

Segurança do RSA

- Actualmente, chaves são normalmente de 1024-2048 bits
- Recomendação é de 2048 bits, pelo menos
  - Chaves de 256 bits quebradas em poucas horas com PC
  - Em 1999, chave de 512 bits foi quebrada por sistema distribuído de centenas de computadores
  - Alguns peritos acreditam que 1024 bits será quebrável a curto-prazo
  - Computador quântico (se algum dia vier a existir) quebra chave RSA facilmente (tempo polinomial)
    - Usando Algoritmo de Shor

A computação quântica?

Comparação cifra assimétrica

- Em 2003, a RSA Security afirma:
  - 1024-bit RSA keys are equivalent in strength to 80-bit symmetric keys
  - 2048-bit RSA keys to 112-bit symmetric keys
  - 3072-bit RSA keys to 128-bit symmetric keys
- RSA afirma:
  - 1024-bit keys quebráveis em 2010
  - 2048-bit keys suficientes até 2030
  - Key length of 3072 bits should be used if security is required beyond 2030
- NIST key management guidelines
  - Suggest that 15360-bit RSA keys are equivalent in strength to 256-bit symmetric keys

Considerações genéricas sobre utilização de algoritmos de criptografia

Distribuição e gestão de chaves

Cifras simétricas

Há que divulgar um valor secreto a um conjunto limitado de interlocutores legítimos, que o devem manter secreto

Cifras assimétricas

Há que garantir que a chave privada apenas é conhecida pela entidade a que pertence

Há que garantir que a chave pública é verdadeira e que não alvo de um ataque de *"man-in-the-middle"*

Cifra híbrida (ou mista)

- Os algoritmos de cifra assimétrica são computacionalmente mais complexos que cifra simétrica
  - 100 a 1000 vezes mais lentos
- Mas a distribuição da chave pública é mais prática que a chave secreta
- Como conseguir o melhor dos dois mundos?
- **Cifras híbridas**
  - Gera-se chave secreta, chamada chave de sessão
  - Usa-se cifra assimétrica para trocar apenas uma chave secreta
  - Usa-se cifra simétrica e a chave secreta para os restantes dados