Criptografia III

Criptografia

António Santos

ISTEC

Tópicos

Cifras de fluxo

Uma cifra de fluxo é aquela que encripta um fluxo de dados digital um bit ou um byte de cada vez. Alguns exemplos de cifras de fluxo clássicas são as autochaveadas Vigenère e Vernam. No caso ideal, uma versão onetime pad da cifra Vernam seria utilizada, na qual o fluxo de chaves (k_i) tem o tamanho do fluxo de bits do texto claro (p_i).

Cifras de bloco

Uma cifra de bloco é aquela em que um bloco de texto claro é tratado como um todo e usado para produzir um de texto cifrado com o mesmo tamanho. Normalmente, um tamanho de bloco de 64 ou 128 bits é utilizado. Assim como a cifra de fluxo, os dois utilizadores compartilham uma chave de encriptação simétrica. Uma cifra de bloco pode ser usada para conseguir o mesmo efeito de uma cifra de fluxo.

cifra de bloco com um bloco de texto claro de $n = 2$ bits

mapeamento reversível	
Texto claro	Texto cifrado
00	11
01	10
10	00
11	01

mapeamento irreversível	
Texto claro	Texto cifrado
00	11
01	10
10	01
11	01

Cifra de substituição

Texto claro	Texto cifrado
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100

Texto claro	Texto cifrado
0000	1110
0001	0011
0010	0011
0011	1000
0100	0001
0101	1100
0110	1010
0111	1111
1000	0111
1001	1101
1010	1001
1011	0110

No caso de $n = 4$

$$
\begin{aligned}
y_1 &= k_{11}x_1 + k_{12}x_2 + k_{13}x_3 + k_{14}x_4 \\
y_2 &= k_{21}x_1 + k_{22}x_2 + k_{23}x_3 + k_{24}x_4 \\
y_3 &= k_{31}x_1 + k_{32}x_2 + k_{33}x_3 + k_{34}x_4 \\
y_4 &= k_{41}x_1 + k_{42}x_2 + k_{43}x_3 + k_{44}x_4
\end{aligned}
$$

Definição

A Cifra de Feistel aproxima a cifra de bloco ideal utilizando o conceito de uma cifra de produto, que é a execução de duas ou mais cifras simples em sequência, de tal forma que o resultado ou produto final seja criptograficamente mais forte do que qualquer uma das cifras componentes. A essência da técnica é desenvolver uma cifra de bloco com um tamanho de chave de k bits e de bloco de n bits, permitindo um total de 2^k transformações possíveis, em vez de $2^n!$ transformações disponíveis com a cifra de bloco ideal.

Cifra de Feistel

Cifra que alterna substituições e permutações

- Substituição: cada elemento de texto claro ou grupo de elementos é substituído exclusivamente por um elemento ou grupo de elementos de texto cifrado correspondente.
- Permutação: uma sequência de elementos de texto claro é substituída por uma permutação dessa sequência.

Difusão e confusão

- Difusão: a estrutura estatística do texto claro é dissipada em estatísticas de longa duração do texto cifrado.
- Confusão: procura-se estabelecer o relacionamento entre as estatísticas do texto cifrado e o valor da chave de encriptação o mais complexo possível, novamente para frustrar tentativas de descobrir a chave.

Difusão e confusão

- Tamanho de bloco: tamanhos de bloco maiores significam maior segurança, mas velocidade de encriptação/decriptação reduzida para determinado algoritmo.
- Tamanho de chave: tamanho de chave maior significa maior segurança, mas pode diminuir a velocidade de encriptação/decriptação.

Difusão e confusão

- Número de rotações: a essência da cifra de Feistel é que uma única rotação oferece segurança inadequada, mas várias proporcionam maior segurança. Um tamanho típico é de 16 rotações.
- Algoritmo de geração de subchave: maior complexidade nesse algoritmo deverá levar a maior dificuldade de criptoanálise.
- Função F: novamente, maior complexidade geralmente significa maior resistência à criptoanálise.

Difusão e confusão

Existem duas considerações no projeto da cifra de Feistel:

- Encriptação e decriptação rápidas em software: em muitos casos, a encriptação é embutida nas aplicações ou funções utilitárias, de tal forma que impede uma implementação em hardware. A velocidade de execução do algoritmo torna-se uma preocupação.
- Facilidade de análise: embora quiséssemos tornar nosso algoritmo o mais difícil possível de criptoanalisar, existe um grande benefício em colocá-lo como fácil de ser analisado. Se o algoritmo puder ser explicado de forma concisa e clara, é mais fácil analisá-lo em busca de vulnerabilidades criptoanalíticas e, portanto, desenvolver um nível mais alto de garantia quanto à sua força.

Algoritmo De Decriptação de Feistel

O processo de decriptação com uma cifra de Feistel é basicamente o mesmo de encriptação. A regra é a seguinte: use o texto cifrado como entrada para o algoritmo, mas as subchaves K_i em ordem reversa. Ou seja, usa-se K_n na primeira rotação, K_{n-1} na segunda, e assim por diante, até K_1 ser usada na última rotação.

Algoritmo De Decriptação de Feistel

Seja $DE7F03A6$. Então, $LE_{14} = DE7F$ e $RE_{14} = 03A6$. Além disso, admita que o valor de K_{15} seja $12DE52$. Após a rotação 15, temos $LE_{15} = 03A6$ e $RE_{15} = F(03A6, 12DE52) \oplus DE7F$.

Decriptação de Feistel

Vamos supor que $LD_1 = RE_{15}$ e $RD_1 = LE_{15}$, então $LD_2 = RE_{14}$ e $RD_2 = LE_{14}$. Assim, começamos com $LD_1 = F(03A6, 12DE52) \oplus DE7F$ e $RD_1 = 03A6$, daqui resulta $LD_2 = 03A6 = RE_{14}$ e $RD_2 = F(03A6, 12DE52) \oplus [F(03A6, 12DE52) \oplus DE7F] = DE7F = LE_{14}$.

Definição

O DES é um algoritmo de criptografia baseado em blocos. Um bloco de tamanho fixo de bit é obtido e transformado por uma série de operações básicas em outro bloco criptografado do mesmo tamanho. No caso do DES, o tamanho do bloco é de 64 bits. A chave também possui 64 bits, mas 8 desses bits são usados para verificar a paridade, tornando o comprimento efetivo da chave em 56 bits.

Metodologia

O DES é composto por 16 fases ou rotações idênticas. No início e no final é realizada uma permutação. Essas permutações não são significativas ao nível da criptografia, pois foram incluídas para facilitar o upload e download dos blocos no hardware dos anos 70.

Metodologia

1. Expansão: pega-se metade do bloco de 64 bits (32 bits) que expandido para 48 bits mediante expansão da permutação duplicando alguns dos bits.
2. Mistura: o resultado é combinado com uma subchave usando uma operação XOR. Dezesseis subchaves (uma para cada rotação) são derivadas da chave inicial, gerando subchaves.
3. Substituição: após a mistura, o bloco é dividido em oito pedaços de 6 bits que são passados para as caixas de substituição.
4. Permutação: finalmente, os 32 bits de saída das S-caixas são reordenados de acordo com uma permutação fixa.

Matriz PC1

$$\begin{bmatrix} 57 & 49 & 41 & 33 & 25 & 17 & 09 & 01 & 58 & 50 & 42 & 34 & 26 & 18 \\ 10 & 02 & 59 & 51 & 43 & 35 & 27 & 19 & 11 & 03 & 60 & 52 & 44 & 36 \\ 63 & 55 & 47 & 39 & 31 & 23 & 15 & 07 & 62 & 54 & 46 & 38 & 30 & 22 \\ 14 & 06 & 61 & 53 & 45 & 37 & 29 & 21 & 13 & 05 & 28 & 20 & 12 & 04 \end{bmatrix}$$

Aplicação da matriz

Se tivéssemos uma chave "M"igual a:

$$M = 0001001100110100010101110111100110011011101111001101111111100$$

Após passa-la pela matriz de permutação e compressão acima teríamos a seguinte sub chave PC1(M) de 56 bits.

$$PC1(M) = 11110000110011001010101011110101010101100110011110001111$$

Rotação

$C_0 = 11110000110011001010101011111$
$D_0 = 01010101011001100111100011111$
C_1 e D_1 que é a rotação de 1 bit a esquerda de C_0 e D_0.
$C_1 = 11100001100110010101011111$
$D_1 = 10101010110011001111100011110$
...

concatenação de C_n com D_n

$C_1D_1 =$
$1110000110011001010101011111101010101100110011110001111 0$
$C_2D_2 =$
$1100001100110010101010111111010101011001100111100011110 1$
$C_3D_3 =$
$0000110011001010101011111111010101100110011110001111010 1$
...

Matriz PC2

$$\begin{bmatrix} 14 & 17 & 11 & 24 & 01 & 05 & 03 & 28 \\ 15 & 06 & 21 & 10 & 23 & 19 & 12 & 04 \\ 26 & 08 & 16 & 07 & 27 & 20 & 13 & 02 \\ 41 & 52 & 31 & 37 & 47 & 55 & 30 & 40 \\ 51 & 45 & 33 & 48 & 44 & 49 & 39 & 56 \\ 34 & 53 & 46 & 42 & 50 & 36 & 29 & 32 \end{bmatrix}$$

$K_n = C_n D_n$

$$K1 = 000110110000001011101111111111000111000001110010$$
$$K2 = 011110011010111011011001110110111100100111100101$$
$$K3 = 010101011111111001000101001000010110011111100110 01$$
$$.....$$

Metodologia

A forma mais simples de se decifrar um bloco cifrado com o DES é simplesmente fazendo o inverso de cifrar. Ou seja, seguindo os mesmos passos anteriormente descritos porém invertendo a ordem das sub-chaves $(K_1, \cdots, K_{16})$

Metodologia

$$P1(M') =$$
$$0000101001001100110110011001010110111100010000100011001000110$$

que em seguida se divide nos blocos:

$$L_0 = 00001010010011001101100110010101$$
$$R_0 = 10111100010000100011001000110100$$

Metodologia

$$L_n = R_{n-1}$$
$$R_n = L_{n-1} \oplus f(R_{n-1}, K_n)$$

Desenvolvimento

$L_n = R_{n-1}$.
$L_1 = 10111100010000100011001000110100$
Para calcular $f(R_0, K_1)$ expandimos R_0 com a matriz PE obtendo $PE(R_0)$.
$PE(R_0) =$
$001000000110101000000100000011010010000011010100 0$
Depois realizamos um XOR entre $PE(R_0)$ e a chave K_{16} que resultará no bloco.
$K_{16} \oplus PE(R_0) =$
$111010110101011110001111000101000101011001011101$

Desenvolvimento

Do mesmo modo seria obtido os demais R_n e L_n até a decima sexta iteração onde seria possível escrever:
$R_{16} = 11001100000000001100110011111111$
$L_{16} = 11110000101010101111000010101010$
Que após concatenados e permutados pela matriz $P3$
resultaria no bloco $M =$
$0000000100100011010001010110011111000100110101011110011011110$
que é a mensagem original.

Exemplo

Texto claro:	02468aceeca86420
Chave:	0f1571c947d9e859
Texto cifrado:	da02ce3a89ecac3b

Exemplo de DES

Rodada	K_i	L_i	R_i
IP		5a005a00	3cf03c0f
1	1e030f03080d2930	3cf03c0f	bad22845
2	0a31293432242318	bad22845	99e9b723
3	23072318201d0c1d	99e9b723	0bae3b9e
4	05261d3824311a20	0bae3b9e	42415649
5	3325340136002c25	42415649	18b3fa41
6	123a2d0d04262a1c	18b3fa41	9616fe23
7	021f120b1c130611	9616fe23	67117cf2
8	1c10372a2832002b	67117cf2	c11bfc09
9	04292a380c341f03	c11bfc09	887fbc6c
10	2703212607280403	887fbc6c	600f7e8b
11	2826390c31261504	600f7e8b	f596506e
12	12071c241a0a0f08	f596506e	738538b8
13	300935393c0d100b	738538b8	c6a62c4e
14	311e09231321182a	c6a62c4e	56b0bd75
15	283d3e0227072528	56b0bd75	75e8fd8f
16	2921080b13143025	75e8fd8f	25896490
IP^{-1}		da02ce3a	89ecac3b

Efeito avalanche no DES: mudança no texto claro.

Rodada		δ
	02468aceeca86420 12468aceeca86420	1
1	3cf03c0fbad22845 3cf03c0fbad32845	1
2	bad2284599e9b723 bad3284539a9b7a3	5
3	99e9b7230bae3b9e 39a9b7a3171cb8b3	18
4	0bae3b9e42415649 171cb8b3ccaca55e	34
5	4241564918b3fa41 ccaca55ed16c3653	37
6	18b3fa419616fe23 d16c3653cf402c68	33
7	9616fe2367117cf2 cf402c682b2cefbc	32
8	67117cf2c11bfc09 2b2cefbc99f91153	33

Rodada		δ
9	c11bfc09887fbc6c 99f911532eed7d94	32
10	887fbc6c600f7e8b 2eed7d94d0f23094	34
11	600f7e8bf596506e d0f23094455da9c4	37
12	f596506e738538b8 455da9c47f6e3cf3	31
13	738538b8c6a62c4e 7f6e3cf34bc1a8d9	29
14	c6a62c4e56b0bd75 4bc1a8d91e07d409	33
15	56b0bd7575e8fd8f 1e07d4091ce2e6dc	31
16	75e8fd8f25896490 1ce2e6dc365e5f59	32
IP^{-1}	da02ce3a89ecac3b 057cde97d7683f2a	32

Efeito avalanche no DES: mudança na chave

Rodada		δ
	02468aceeca86420 02468aceeca86420	0
1	3cf03c0fbad22845 3cf03c0f9ad628c5	3
2	bad2284599e9b723 9ad628c59939136b	11
3	99e9b7230bae3b9e 9939136b768067b7	25
4	0bae3b9e42415649 768067b75a8807c5	29
5	4241564918b3fa41 5a8807c5488dbe94	26
6	18b3fa419616fe23 488dbe94aba7fe53	26
7	9616fe2367117cf2 aba7fe53177d21e4	27
8	67117cf2c11bfc09 177d21e4548f1de4	32

Rodada		δ
9	c11bfc09887fbc6c 548f1de471f64dfd	34
10	887fbc6c600f7e8b 71f64dfd4279876c	36
11	600f7e8bf596506e 4279876c399fdc0d	32
12	f596506e738538b8 399fdc0d6d208dbb	28
13	738538b8c6a62c4e 6d208dbbb9bdeeaa	33
14	c6a62c4e56b0bd75 b9bdeeaad2c3a56f	30
15	56b0bd7575e8fd8f d2c3a56f2765c1fb	33
16	75e8fd8f25896490 2765c1fb01263dc4	30
IP^{-1}	da02ce3a89ecac3b ee92b50606b62b0b	30

Definição

O Advanced Encryption Standard (Padrão de Encriptação Avançada), ou AES, foi publicado pelo National Institute of Standards and Technology (NIST), em 2001. O AES é uma cifra simétrica de bloco que pretende substituir o DES como o padrão para uma grande variedade de aplicações. Comparada às cifras de chave pública como a RSA, a estrutura do AES e a maioria das cifras simétricas são bastante complexas e não explicadas tão facilmente quanto outras cifras criptográficas.

Aritmética

No AES, todas as operações são realizadas com 8 bits. Em particular, as operações aritméticas de soma, multiplicação e divisão são feitas sobre o corpo finito $GF(2^8)$.
Basicamente, um corpo é um conjunto no qual nós podemos somar, subtrair, multiplicar e dividir sem sair dele. A divisão é definida com a seguinte regra: $a/b = a(b^{-1})$. Um exemplo de um corpo finito (aquele que possui um número finito de elementos) é o conjunto $\mathbb{Z}_p$ que contém todos os inteiros $\{0, 1, \cdots, p-1\}$, onde p é um número primo e cujo cálculo é feito módulo p.

Corpo finito

Corpo finito contendo 2^n elementos, sendo esse corpo denominado $GF(2^n)$. Considere o conjunto S de todos os polinómios de grau $n-1$ ou menor com coeficientes binários. Então, cada polinómio possui a forma

$$f(x) = a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_0 = \sum_{j=0}^{n-1} a_i x^i$$

onde cada a_i assume o valor 0 ou 1.

Estrutura geral do AES

- Um recurso digno de nota é que ela não é uma estrutura Feistel. Lembre-se de que, na estrutura Feistel clássica, metade do bloco de dados é usada para modificar a outra metade, e depois elas são invertidas. Em vez disso, AES processa o bloco de dados inteiro como uma única matriz durante cada rotação usando substituições e permutação.
- A chave que é fornecida como entrada é expandida para um array de quarenta e quatro palavras (words) de 32 bits cada um, $w[i]$. Quatro palavras (words) distintas (128 bits) servem como uma chave para cada rotação.

Estrutura geral do AES

- Quatro estágios diferentes são usados, um de permutação e três de substituição:
  - SubBytes: utiliza uma S-box para realizar uma substituição byte a byte do bloco
  - ShiftRows: uma permutação simples
  - MixColumns: uma substituição que utiliza aritmética sobre $GF(2^8)$
  - AddRoundKey: um XOR bit a bit simples do bloco atual com uma parte da chave expandida

Estrutura geral do AES

- A estrutura é muito simples. Para a encriptação e decriptação, a cifra começa com um estágio AddRoundKey, seguido por nove rotações, e cada uma inclui todos os quatro estágios, seguidas por uma décima rotação de três estágios.
- O estágio AddRoundKey utiliza a chave. Por esse motivo, a cifra começa e termina com ele. Qualquer outro estágio, aplicado no início ou no fim, é reversível sem conhecimento da chave e não impacta na segurança.
- O estágio AddRoundKey, com efeito, é uma forma de cifra de Vernam. Os outros três estágios oferecem confusão, difusão e não linearidade, mas isolados não ofereceriam segurança, pois não usam a chave.

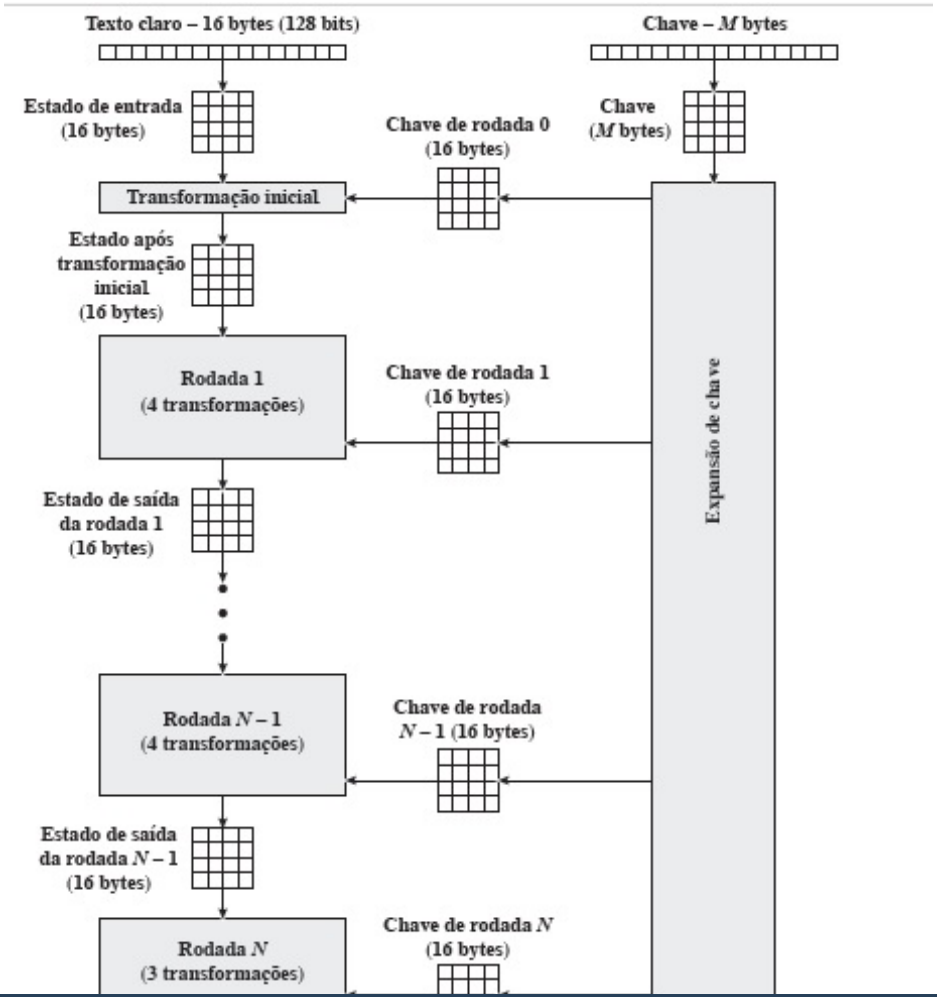
Estrutura geral do AES

- Cada estágio é facilmente reversível. Para os estágios SubByte, ShiftRows e MixColumns, uma função inversa é usada no algoritmo de decriptação. Para o AddRoundKey, o inverso é obtido pelo XOR da mesma chave da rotação com o bloco, usando o resultado de que $A \oplus B \oplus B = A$.
- Assim como na maioria das cifras em bloco, o algoritmo de decriptação utiliza a chave expandida em ordem reversa. Porém, o algoritmo de decriptação não é idêntico ao de encriptação. Consequência da estrutura do AES.

Estrutura geral do AES

- Uma vez estabelecido que os 4 estágios são reversíveis, verifica-se que a decriptação recupera o texto claro.
- rotação final da encriptação e da decriptação consiste em apenas três estágios. Consequência da estrutura do AES, necessário para tornar a cifra reversível.

Processo de encriptação do AES



Comprimento da chave

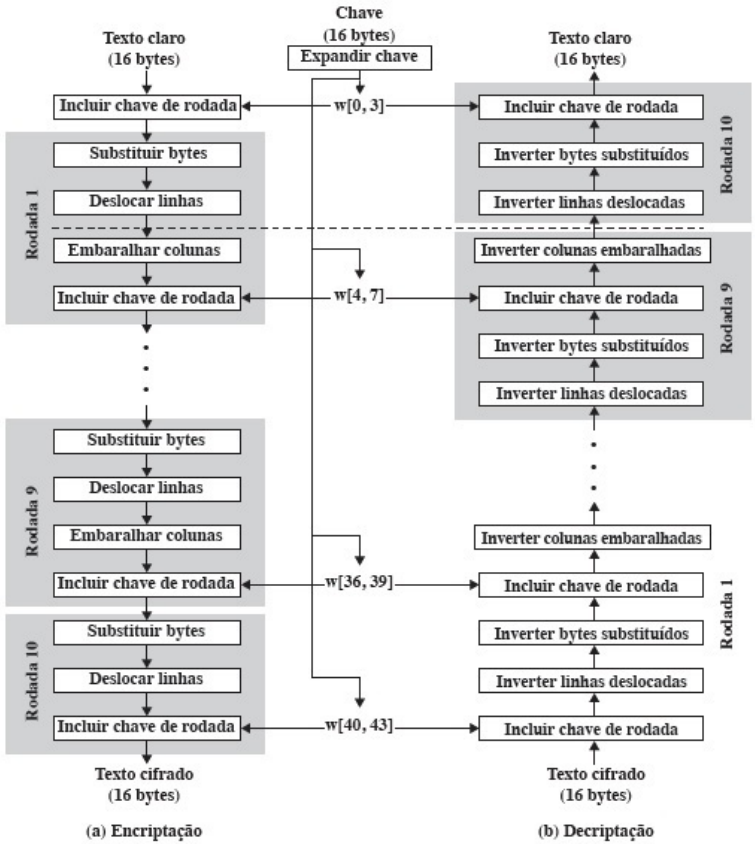
O comprimento da chave pode ser 16, 24 ou 32 bytes (128, 192 ou 256 bits). O algoritmo é denominado AES-128, AES-192 ou AES-256, dependendo do tamanho da chave.
A entrada para os algoritmos de encriptação e decriptação é um único bloco de 128 bits.

Parâmetros do AES

Tamanho da chave (words/bytes/bits)	4/16/128	6/24/192	8/32/256
Tamanho do bloco de texto claro (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Número de rodadas	10	12	14
Tamanho da chave de rodada (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Tamanho da chave expandida (words/bytes)	44/176	52/208	60/240

Estrutura Detalhada

Encriptação e Desencriptação no AES.



Exemplo

Texto claro:	0123456789abcdeffedcba9876543210
Chave:	0f1571c947d9e8590cb7add6af7f6798
Texto cifrado:	ff0b844a0853bf7c6934ab4364148fb9

A força das Chaves

Tamanho de chave (bits)	Cifra	Número de chaves alternativas	Tempo exigido a 10^9 decriptações/s	Tempo exigido a 10^{13} decriptações/s
56	DES	$2^{56} \approx 7{,}2 \times 10^{16}$	2^{55} ns = 1,125 ano	1 hora
128	AES	$2^{128} \approx 3{,}4 \times 10^{38}$	2^{127} ns = $5{,}3 \times 10^{21}$ anos	$5{,}3 \times 10^{17}$ anos
168	Triple DES	$2^{168} \approx 3{,}7 \times 10^{50}$	2^{167} ns = $5{,}8 \times 10^{33}$ anos	$5{,}8 \times 10^{29}$ anos
192	AES	$2^{192} \approx 6{,}3 \times 10^{57}$	2^{191} ns = $9{,}8 \times 10^{40}$ anos	$9{,}8 \times 10^{36}$ anos
256	AES	$2^{256} \approx 1{,}2 \times 10^{77}$	2^{255} ns = $1{,}8 \times 10^{60}$ anos	$1{,}8 \times 10^{56}$ ano
26 caracteres (permutação)	Monoalfabético	$2! = 4 \times 10^{26}$	2×10^{26} ns = $6{,}3 \times 10^{9}$ anos	$6{,}3 \times 10^{6}$ anos

Figura: Tempo médio exigido para uma busca exaustiva no espaço de chaves.