Criptografia

Prof. Ricardo S. Puttini, MSc
Departamento de Engenharia Elétrica
Universidade de Brasília

Conteúdo

- Princípios de Teoria da Informação
- Introdução aos Sistemas Criptográficos
- Criptografia Simétrica
- Criptografia Assimétrica
- Funções de Condensação (Hash Functions)
- Infra-estrutura de Chaves Públicas (ICP)

Princípios de Teoria da Informação

- O problema da codificação consiste na busca de formas eficientes de representação da informação:
  - Economia de símbolo utilizados (compressão);
  - Proteção contra erros (detecção/correção de erros);
  - Serviços de segurança da informação (criptografia).
- Teoria da Informação: construção de modelos matemáticos para **linguagem**.
- Linguagem: conjunto de características da informação que se deseja codificar.

Quantidade de Informação

- Teoria de Shannon (1948): modelo probabilístico para a linguagem.
- Quantidade de Informação

$$I(E) = -log_2 p(E)$$

- A base 2 do logaritmo permite que a quantidade de informação seja medida em bits
- Exemplos:
  - E1 = “o sol nascerá amanhã”;
    $p(E1) = 1 \rightarrow I(E1) = 0$
  - E2 = “ao lançar-se a moeda, obteve-se cara”;
    $p(E2) = 0,5 \rightarrow I(E2) = 1$ (bit)

Princípios de Teoria da Informação

- Exemplo: Considere uma fonte que gere símbolos 0's e 1's de forma independente e equiprováveis. Então, a quantidade de informação obtida pelo conhecimento de "n" bits de uma seqüência gerada por essa fonte é:

$$I(S_n) = -\log_2\left(\frac{1}{2^n}\right) = n$$

- Isto é, a quantidade de informação contida em uma seqüência de n bits é igual n bits.

Entropia

- Entropia é a quantidade de informação contida em um evento qualquer, i.e. a quantidade de informação média de um evento típico.
- É uma medida da incerteza associada a uma ocorrência de um evento: quanto maior a entropia, maior a incerteza associada ao evento.
- Para uma v.a. X, que pode assumir um número finito de valores com probabilidades p_1, p_2, ..., p_n, tem-se a seguinte definição de entropia:

$$H(X) = -\sum_{i=1}^{n} p_i \log_2(p_i)$$

Entropia

- Entropia conjunta de um vetor X = [A,B], onde A e B são v.a. assumindo um número finito de valores:

$$H(X) = H(A,B) = -\sum_{i,j} p(A=a_i, B=b_j) \log_2 p(A=a_i, B=b_j)$$

- Note que:

$$H(A,B) \leq H(A) + H(B)$$

Entropia

- Dado duas v.a. X e Y, a incerteza acerca de X, dado que uma ocorrência particular de Y é conhecida pode ser expressa como:

$$H(X/Y=y) = -\sum_{i} p(X=X_i, Y=y) \log_2 p(X=X_i, Y=y)$$

- Define-se entropia condicional, H(X/Y), ou equivocação de X dado Y por:

$$H(X/Y) = -\sum_{j} H(X/Y=y_j) p(Y=y_j)$$

- Assim, a entropia condicional representa a incerteza em relação a uma ocorrência de X, quando uma ocorrência de Y é conhecida.

Propriedades da Entropia

I) $H(X) = -\sum_{i=1}^{n} p_i \log_2(p_i) \leq \log_2(n)$

II) $H(X/Y) = H(X,Y) - H(Y)$

$H(Y/X) = H(X,Y) - H(X)$

III) Se X e Y são v.a. independentes:

$H(X/Y) = H(X)$

$H(Y/X) = H(Y)$

$H(X,Y) = H(X) + H(Y)$

Informação Mútua

- É a quantidade de informação que a ocorrência de um evento Y = y fornece a respeito da ocorrência de um evento X = x:

$$I(x,y) = I(x) - I(x/y) =$$
$$-\log_2 p(X=x) - [-\log_2 p(X=x/Y=y)]$$

- O valor esperado de I(x,y) é definido como a informação mútua entre X e Y:

$$I(X,Y) = \sum_{i,j} p(X=x_i/Y=y_j) I(x_i,y_j)$$

- Note que:

$$I(X,Y) = H(X) + H(Y) - H(X,Y)$$

Modelos de Linguagens e Fontes de Texto

- Associa-se um alfabeto de m-letras,A = $\{a_0, a_1, a_{n-1}\}$, ao conjunto dos inteiros módulo m:

  $a_i \leftrightarrow i; i \in \mathbf{Z_m} = \{0,1,..., m\}$

- Conjunto de todos os n-gramas de A, $\mathbf{Z_m^n}$, forma um novo alfabeto de m^n símbolos.
- Texto: um n-grama sobre um alfabeto $\mathbf{Z_m}$.
- Fontes de texto: modelo capaz de produzir textos pertencentes a uma determinada linguagem.

Modelos de Linguagens e Fontes de Texto

- Fonte de texto de primeira ordem: considera a freqüência relativa dos caracteres de uma linguagem.
  - Não leva em conta a dependência inter-símbolos.
- Fonte de texto de segunda ordem: considera a freqüência relativa de digramas.
- Fonte de texto de ordem superior: considera a freqüência relativa de n-gramas.
- Fontes Markovianas: considera as probabilidades de transição inter-simbolos.

Distribuição de Probabilidades de 1-gramas em Português

Letra - p	Letra - p	Letra - p	Letra - p
A 0,114	H 0,006	O 0,080	V 0,012
B 0,008	I 0,056	P 0,024	W 0,000
C 0,045	J 0,002	Q 0,007	X 0,002
D 0,046	K 0,001	R 0,053	Y 0,0000
E 0,114	L 0,020	S 0,071	Z 0,003
F 0,010	M 0,042	T 0,046	_ 0,156
G 0,011	N 0,041	U 0,030	

Distribuição de Probabilidades de 2-gramas em Português (1/2)

*	A	B	C	D	E	F	G	H	I	J	K	L	M
A	0.00	0.12	0.69	0.88	0.00	0.11	0.29	0.00	0.21	0.00	0.00	0.63	0.46
B	0.02	0.00	0.00	0.00	0.13	0.00	0.00	0.00	0.13	0.01	0.00	0.15	0.00
C	1.46	0.00	0.00	0.00	0.57	0.00	0.00	0.41	0.64	0.00	0.00	0.00	0.00
D	0.90	0.00	0.00	0.00	2.14	0.00	0.00	0.00	0.91	0.00	0.00	0.00	0.00
E	0.08	0.01	0.69	0.09	0.00	0.05	0.28	0.00	0.16	0.12	0.00	0.56	1.22
F	0.13	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.27	0.00	0.00	0.00	0.00
G	0.02	0.00	0.00	0.00	0.52	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
H	0.42	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
I	0.52	0.03	0.73	0.47	0.04	0.25	0.17	0.00	0.00	0.00	0.00	0.33	0.59
J	0.14	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
K	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
L	0.25	0.00	0.01	0.00	0.28	0.04	0.08	0.00	0.43	0.00	0.00	0.00	0.07
M	1.15	0.02	0.00	0.00	0.94	0.00	0.00	0.00	0.17	0.00	0.00	0.00	0.00
N	0.61	0.00	0.40	0.52	0.13	0.09	0.02	0.11	0.20	0.02	0.02	0.00	0.00
O	0.06	0.16	0.23	0.22	0.13	0.03	0.20	0.00	0.04	0.01	0.00	0.13	0.42
P	0.79	0.00	0.00	0.00	0.41	0.00	0.00	0.00	0.03	0.00	0.00	0.17	0.00
Q	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R	1.50	0.02	0.05	0.04	0.81	0.00	0.01	0.00	0.71	0.00	0.00	0.01	0.33
S	0.58	0.01	0.05	0.01	1.19	0.03	0.00	0.00	0.73	0.00	0.00	0.00	0.09
T	0.85	0.00	0.07	0.00	1.46	0.00	0.00	0.00	0.71	0.00	0.00	0.00	0.06
U	0.28	0.12	0.03	0.03	0.37	0.00	0.00	0.00	0.21	0.01	0.00	0.09	0.60
V	0.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.16	0.00	0.00	0.00	0.00
W	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
X	0.02	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00
Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Z	0.27	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00
_	1.09	0.22	1.80	2.32	1.35	0.36	0.13	0.04	0.38	0.02	0.02	0.14	0.56

Distribuição de Probabilidades de 2-gramas em Português (2/2)

	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	_
A	0.65	0.95	0.10	0.03	0.91	1.11	0.75	0.04	0.47	0.01	0.00	0.00	0.05	3.74
B	0.00	0.04	0.00	0.00	0.03	0.08	0.05	0.04	0.00	0.00	0.00	0.00	0.00	0.03
C	0.03	1.15	0.00	0.00	0.24	0.00	0.05	0.13	0.00	0.00	0.00	0.00	0.00	0.02
D	0.00	1.10	0.00	0.00	0.01	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.07
E	1.56	0.01	0.00	0.11	1.08	1.39	0.39	0.03	0.14	0.00	0.13	0.00	0.04	3.56
F	0.00	0.26	0.00	0.00	0.01	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.07
G	0.01	0.11	0.00	0.00	0.01	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00
H	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00
I	0.60	0.31	0.26	0.02	0.15	0.83	0.35	0.02	0.23	0.00	0.01	0.00	0.22	0.08
J	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
K	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
L	0.01	0.39	0.00	0.00	0.00	0.01	0.00	0.05	0.05	0.00	0.00	0.00	0.00	0.13
M	0.00	0.01	0.03	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.13
N	0.00	0.20	0.00	0.00	0.00	0.00	0.00	1.14	0.05	0.00	0.00	0.00	0.00	0.04
O	0.59	0.00	0.13	0.00	0.00	1.07	0.13	0.15	0.00	0.00	0.01	0.00	0.00	3.13
P	0.00	0.59	0.00	0.00	0.02	0.01	0.18	0.10	0.00	0.00	0.00	0.00	0.00	0.01
Q	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00
R	0.05	0.56	0.00	0.01	0.10	0.09	0.15	0.02	0.03	0.00	0.00	0.00	0.00	0.75
S	0.00	0.34	0.04	0.01	0.00	0.00	0.04	0.02	0.01	0.00	0.00	0.00	0.00	2.53
T	0.00	0.78	0.00	0.00	0.46	0.05	0.00	0.13	0.00	0.00	0.00	0.00	0.00	0.03
U	0.14	0.60	0.05	0.00	0.53	0.15	0.33	0.00	0.00	0.00	0.00	0.00	0.01	0.14
V	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
W	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
X	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Z	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04
	0.48	0.68	1.31	0.04	0.31	1.56	0.55	0.29	0.13	0.00	0.01	0.00	0.00	0.00

Probabilidades de Transição para o Português, em % (1/2)

*	A	B	C	D	E	F	G	H	I	J	K	L	M
A	0.00	1.01	6.05	7.74	0.00	0.96	2.56	0.01	1.85	0.03	0.00	0.72	4.07
B	8.72	0.00	0.39	0.08	16.28	0.00	0.00	0.00	16.28	0.78	0.00	19.38	0.39
C	32.45	0.00	0.00	0.00	7.03	0.00	0.00	8.91	14.19	0.00	0.13	1.38	0.00
D	20.95	0.00	0.03	0.00	46.63	0.00	0.00	0.00	5.64	0.00	0.00	0.00	0.07
E	0.72	0.11	4.27	0.82	0.04	0.48	2.43	0.01	1.40	1.01	0.00	3.13	10.72
F	19.22	0.00	0.00	0.00	8.40	0.00	0.00	0.00	27.84	0.00	0.00	0.00	0.00
G	6.09	0.00	0.00	0.00	32.05	0.00	0.00	0.00	9.61	0.00	0.00	0.26	0.13
H	75.00	0.00	0.00	0.00	12.85	0.00	0.00	0.00	3.64	0.00	0.00	0.00	0.00
I	6.48	1.13	13.00	9.31	0.65	4.44	2.13	0.00	0.00	0.00	0.00	4.17	0.11
J	68.70	0.00	0.00	0.00	15.27	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
K	0.00	0.00	0.00	0.00	10.34	0.00	0.00	0.00	5.93	0.00	0.00	0.00	0.00
L	12.62	0.00	0.39	0.00	14.32	2.17	4.33	3.17	22.99	0.00	0.00	0.16	3.79
M	27.49	1.73	0.04	0.04	22.93	0.00	0.00	0.00	2.75	0.00	0.00	0.04	0.00
N	14.72	0.04	9.69	12.67	9.74	2.24	0.96	2.68	4.85	0.09	0.06	0.11	0.00
O	0.78	2.03	2.86	2.77	1.82	0.32	2.45	0.00	0.47	0.15	0.00	1.57	5.83
P	18.38	0.13	0.13	0.00	17.40	0.00	0.00	0.06	1.15	0.00	0.00	5.16	0.00
Q	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
R	28.35	0.00	1.00	0.72	15.27	0.08	0.36	0.00	13.64	0.08	0.00	0.11	6.84
S	6.70	0.08	0.86	0.11	18.72	0.48	0.04	0.04	10.27	0.03	0.00	0.06	1.30
T	18.40	0.00	0.33	0.00	31.75	0.00	0.00	0.00	15.59	0.00	0.00	0.00	1.32
U	0.00	4.04	0.90	0.85	16.66	0.05	0.05	0.00	8.84	0.40	0.00	2.64	18.57
V	12.68	0.00	0.00	0.00	68.29	0.00	0.00	0.00	12.85	0.00	0.00	0.00	0.00
W	68.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
X	14.95	0.00	4.87	0.00	25.23	0.00	0.00	0.00	23.38	0.00	0.00	0.00	0.00
Y	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	11.11	0.00	0.00	0.00	0.00
Z	67.44	0.00	0.00	0.00	14.42	0.00	0.00	0.00	5.12	0.00	0.00	0.00	0.00
	10.88	1.43	11.53	14.08	8.68	2.30	0.96	0.27	2.47	0.15	0.10	0.91	3.57

Probabilidades de Transição para o Português, em % (2/2)

	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
A	5.68	8.20	1.36	0.51	7.05	9.70	2.03	0.58	4.06	0.00	0.04	0.03	0.48	29.66
B	0.19	5.93	0.00	0.00	4.46	10.86	6.58	5.23	0.58	0.00	0.00	0.00	0.03	4.28
C	0.71	25.42	0.00	0.00	5.31	0.00	1.18	2.69	0.00	0.00	0.00	0.03	0.00	0.44
D	0.00	23.84	0.00	0.00	0.30	0.10	0.00	1.19	0.00	0.00	0.00	0.00	0.00	0.17
E	13.86	0.08	0.53	0.84	9.49	12.21	3.24	0.27	1.25	0.00	1.15	0.00	0.32	31.55
F	0.00	28.54	0.18	0.00	20.88	0.00	0.78	1.71	0.00	0.00	0.00	0.00	0.00	0.82
G	0.79	9.64	0.00	0.00	16.28	0.00	0.00	24.24	0.00	0.00	0.00	0.00	0.00	0.00
H	0.00	5.08	0.00	0.00	0.00	0.00	0.00	2.91	0.00	0.00	0.00	0.00	0.00	0.00
I	10.67	5.04	4.63	0.03	2.69	14.66	5.69	0.08	4.06	0.00	0.19	0.00	3.85	1.35
J	0.00	1.03	0.00	0.00	0.00	0.00	0.00	13.74	0.00	0.00	0.00	0.00	0.00	0.75
K	3.45	0.00	6.90	0.00	0.00	6.90	0.00	6.90	0.00	0.00	0.00	0.00	0.00	58.62
L	0.39	9.69	0.00	1.93	0.00	0.31	2.63	2.32	2.09	0.00	0.00	0.00	0.08	16.72
M	0.04	9.68	5.03	0.00	0.00	0.29	0.00	2.10	0.00	0.00	0.00	0.00	0.00	27.93
N	0.00	4.74	0.00	0.15	0.33	10.46	27.68	1.35	1.95	0.00	0.00	0.00	0.00	0.85
O	7.37	0.02	1.67	0.00	10.78	12.70	1.58	1.58	0.48	0.02	0.05	0.00	0.00	42.12
P	0.00	24.85	0.00	0.00	22.01	0.32	7.48	4.41	0.00	0.00	0.00	0.00	0.00	0.35
Q	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
R	0.97	10.55	0.03	0.26	1.89	1.83	2.75	0.40	0.57	0.00	0.00	0.03	0.00	14.89
S	0.02	4.78	2.00	0.15	0.00	5.82	13.32	3.09	0.11	0.02	0.00	0.00	0.00	35.98
T	0.00	17.05	0.00	0.00	10.00	1.15	0.03	3.63	0.00	0.10	0.00	0.10	0.00	0.58
U	4.74	0.15	1.05	0.00	9.36	4.44	10.93	0.00	0.00	0.00	0.00	0.00	0.45	4.09
V	0.00	6.24	0.00	0.00	0.87	0.00	0.00	0.52	0.00	0.00	0.00	0.00	0.00	0.12
W	16.67	16.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
X	0.00	4.67	9.41	0.00	0.00	0.00	10.89	0.00	0.00	0.00	0.00	0.00	0.00	2.80
Y	22.22	0.00	11.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	55.56
Z	0.00	0.47	0.00	0.00	0.00	0.47	0.00	0.00	0.00	0.00	0.00	0.00	0.00	11.16
_	3.10	5.68	8.43	3.12	1.99	10.00	3.50	5.08	1.07	0.00	0.04	0.03	0.03	0.00

Entropia e Redundância de Fontes de Texto

- Uma vez admitido o modelo estatístico de uma linguagem, pode-se calcular a entropia da linguagem
  - Entropia, usando um modelo de fonte de texto de primeira ordem:

$$H_{portugues} \leq H^1_{portugues} = -\sum_i p_i \log_2 p_i = 3{,}97 \text{bits/letra}$$

  - Entropia, usando um modelo de fonte de texto de segunda ordem:

$$H_{portugues} \leq H^2_{portugues} = -\frac{1}{2}\sum_{i,j} p(i,j) \log_2 p(i,j) = 3{,}53 \text{bits/letra}$$

Entropia do Inglês (Shannon)

Entropia do Inglês (Shannon).

	Z_{26}	Z_{27}
H_E^0	4.70	4.76
H_E^1	4.14	4.03
H_E^2	3.56	3.32
H_E^3	3.50	3.10

Entr. do Inglês - Mod. Markoviano Exp. (Shannon).

n	Limite Inferior	Limite Superior
1	3.2	4.0
2	2.5	3.4
3	2.1	3.0
4	1.7	2.6
5	1.7	2.7
10	1.0	2.1
100	0.6	1.3

Entropia e Redundância de Fontes de Texto

- Dada uma fonte sobre um alfabeto $\mathbf{Z_m}$ e entropia **H**, pode-se codificar textos produzidos por essa fonte em um código compacto, onde um n-grama típico pode ser codificado com $\lambda(n)$ letras de $\mathbf{Z_m}$.
- $\lambda(n)$ é o comprimento médio dos códigos necessário para representar um n-grama.

$$\lambda(n) = \frac{nH}{\log_2 m}$$

Entropia e Redundância de Fontes de Texto

- Define-se redundância de uma fonte como a diferença entre o número de símbolos utilizados em um n-grama, obviamente n, e o número médio mínimo de símbolos λ(n) que poderiam ser utilizados por uma codificação mais apropriada:

$$R = 1 - \frac{H}{\log_2 m}$$

- Estima-se normalmente que a redundância da língua inglesa seja de, aproximadamente, 40%.
- Shannon mostrou que, considerando-se efeitos mais amplos, a entropia do inglês se reduziria a 1bit por letra, o que equivale a uma redundância de cerca de 75%.

Introdução aos Sistemas Criptográficos

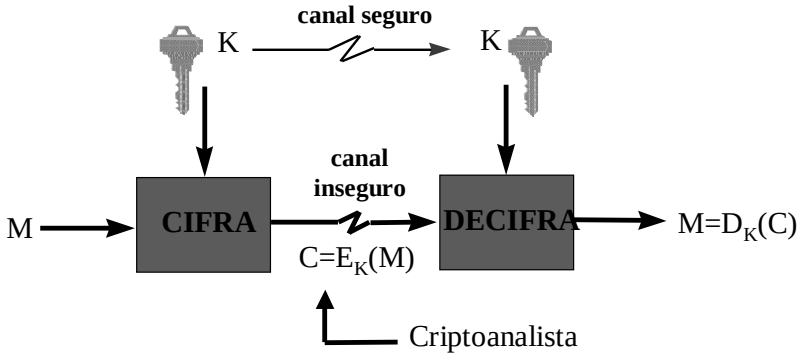
- Criptologia: estudo dos processos criptográficos.
- Formada por duas disciplina:
  - Criptografia: como projetar e construir de sistemas criptográficos
  - Criptoanálise: como “quebrar” sistemas criptográficos.

Introdução aos Sistemas Criptográficos

- Serviços de segurança:
  - Confidencialidade: confiança em que a mensagem possa ser lida apenas pelo receptor desejado
  - Autenticação da Origem: garantia sobre quem originou a mensagem
  - Integridade: confiança em que a mensagem não tenha sido modificada desde o momento da criação
  - Não-retratação: quem originou a mensagem não pode negar tê-lo feito.

Introdução aos Sistemas Criptográficos

- Sistema Criptográfico Convencional



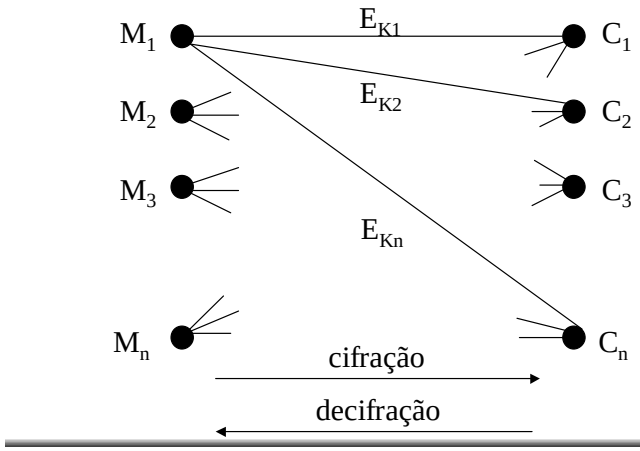
Modelo Estatístico de um Sistema Criptográfico

- Uma mensagem (M) é um n-grama de um alfabeto ***A***
- O processo de cifração de m gera um criptograma (C), designado por c = E(M)
- Claramente, deve haver D(C) = D (E(M)) = m
- O processo de cifração pode ser visto como um conjunto de transformações $\boldsymbol{E}=\{E^n, 1\leq n<\infty\}$, $\boldsymbol{E}^n: \boldsymbol{Z}_m{}^n \rightarrow \boldsymbol{Z}_m{}^n$

Modelo Estatístico de um Sistema Criptográfico

- Sistemas Criptográficos
  - Família de transformações criptográficas $\boldsymbol{E}=\{E_K, K\in \boldsymbol{K}\}$, onde cada transformação é indexada por um parâmetro k, denominado chave criptográfica.
  - Composto de uma tripla (***M***,***K***,***C***)
    - ***M***: espaço das mensagens
    - ***K***: espaço das chaves
    - ***C***: espaço dos criptogramas

Modelo Estatístico de um Sistema Criptográfico



Modelo Estatístico de um Sistema Criptográfico

- Criptoanálise
  - Supõe-se sempre que o criptoanalista conhece
    - Algoritmo;
    - Linguagem do texto em claro;
    - Contexto das mensagens (algumas mensagens são conhecidas).
  - Ataques contra algoritmos criptográficos
    - Ataque com mensagem conhecida;
    - Ataque com mensagem escolhida;
    - Ataque com criptograma escolhido;
    - Ataque ao criptograma apenas.
  - Segurança do algoritmo criptográfico:
    - No mínimo, deve suportar ataques com mensagens conhecidas.
    - Em geral, deve suportar ataques com mensagens/criptograma escolhido.

Modelo Estatístico de um Sistema Criptográfico

- Cifra de César (substituição simples):
  - Cifração: $C_i = E(K,M_i) = (M_i + K)$ *mod* m
  - Decifração: $M_i = D(K,C_i) = (C_i - K)$ *mod* m
- Exemplo: K=7, m = 27 (27 letras + espaço)

  `M = TROPAS`

  `C = _YVWHZ`
- Ataque com mensagem conhecida...
- Ataque com mensagem escolhida...

Modelo Estatístico de um Sistema Criptográfico

- Substituição mono-alfabética: cada letra do alfabeto da mensagem corresponde a uma letra do alfabeto do criptograma:
  - chave = correspondência entre letras
  - Exemplo: `ABCDEFGHIJKLMNOPQRSTUVWVXYZ_`
    `HZEVGVAJXK_MPNQFROIUBWSTYCLD`
- Ataque com mensagem conhecida: qual o tamanho da mensagem para permitir revelar a chave completa?
- Ataque com mensagem escolhida...

Modelo Estatístico de um Sistema Criptográfico

- Dado um modelo estatístico para a linguagem que será a fonte de mensagens, as distribuições de probabilidade P(M) e P(K) permitem obter:

$$P(C) = \sum_{M,K;C=E_K(M)} P(M)P(K) \qquad P(M,K) = P(M)P(K)$$

$$P(M,C) = \sum_{K;C=E_K(M)} P(M)P(K) \qquad P(M/C) = \frac{P(M,C)}{P(C)}$$

$$P(K,C) = \sum_{M;C=E_K(M)} P(M)P(K) \qquad P(K/C) = \frac{P(K,C)}{P(C)}$$

Modelo Estatístico de um Sistema Criptográfico

- Função de Decisão: função que faça $\delta(C) = M$, se $C = E_K(M)$.
- Esta função deverá maximizar p(M/C)

Modelo Estatístico de um Sistema Criptográfico

- Ex1 - Cifra de César:

  K=7, m = 27

  M = TROPAS

  C = _YVWHZ

		Modelo p(M/C)		
K	M = (C - K)	Ordem1	Ordem2	Markov-1
0	YVWHZ	4.779E-16	0.000E+00	0.000E+00
...				
5	VTQBCU	3.164E-10	0.000E+00	0.000E+00
6	USPQST	1.139E-10	0.000E+00	0.000E+00
7	TROPAS	8.051E-08	**8.864E-08**	**1.397E-07**
8	SQNO_R	1.528E-08	6.148E-10	0.000E+00
...				
20	GSBCOF	3.872E-10	2.912E-11	1.256E-11
21	FDABNE	2.027E-09	0.000E+00	0.000E+00
22	EC_AMD	**1.908E-07**	1.174E-09	3.440E-11
23	DBC_LC	1.749E-10	0.000E+00	0.000E+00
...				

Modelo Estatístico de um Sistema Criptográfico

Modelo	S(C_YVWHZ)	K
1ª Ordem	EC_AMD	22
2ª Ordem	TROPAS	7
Markov 1ª Ordem	TROPAS	7

- Obs1: o processo não destrói as estatísticas de n-gramas da linguagem do texto claro.
- Obs2: o espaço de chaves é pequeno, tornando um ataque por força bruta possível.

Modelo Estatístico de um Sistema Criptográfico

- Ex2: M = MANDE_TROPAS
  C = THUKLG_YVWHZ

		Modelo - p(M/C)		
K	M	Ordem-1	Ordem-2	Markov-1
0	THUKLG_YVWHZ	4.086E-28	0.000E+00	0.000E+00
1	SGTJKFZXUVGY	1.133E-27	0.000E+00	0.000E+00
2	RFSIJEYWTUFX	1.500E-25	0.000E+00	0.000E+00
3	QERHIDXVSTEW	5.498E-22	0.000E+00	0.000E+00
4	PDQGHCWURSDV	1.718E-22	0.000E+00	0.000E+00
5	OCPFGBVTQRCU	2.548E-20	0.000E+00	0.000E+00
6	NBOEFAUSPQBT	5.089E-19	0.000E+00	0.000E+00
7	MANDE_TROPAS	**7.049E-15**	**1.389E-13**	**1.213E-12**
8	L_MCDZSQNO_R	1.418E-17	0.000E+00	0.000E+00
9	KZLBCYRPMNZQ	8.078E-26	0.000E+00	0.000E+00
10	JYKABXQOLMYP	3.089E-28	0.000E+00	0.000E+00
11	IXJ_AWPNKLXO	3.744E-25	0.000E+00	0.000E+00
12	HWIZ_VQMJKWN	2.537E-27	0.000E+00	0.000E+00
13	GVHYZUNLIJVM	6.308E-25	0.000E+00	0.000E+00
14	FUGXYTMKHIUL	1.554E-25	0.000E+00	0.000E+00
15	ETFWXSLJGHTK	3.518E-26	0.000E+00	0.000E+00
16	DSEVWRKJFGSJ	9.948E-24	0.000E+00	0.000E+00
17	CRDUVQJHEFRI	1.403E-20	0.000E+00	0.000E+00
18	BQCTUPIGDEQH	1.498E-20	0.000E+00	0.000E+00
19	APBSTOHFCDPG	2.205E-19	0.000E+00	0.000E+00
20	_OARSNGEBCOF	9.278E-17	2.858E-20	1.057E-22
21	ZN_QRMFDABNE	7.391E-19	0.000E+00	0.000E+00
22	YMZPQLEC_AMD	1.285E-20	0.000E+00	0.000E+00
23	XLYOPKDBZ_LC	6.786E-25	0.000E+00	0.000E+00
24	WKXNOJCAYZKB	3.901E-30	0.000E+00	0.000E+00
25	VJWMNIB_XYJA	1.528E-26	0.000E+00	0.000E+00
26	UIVLMHAZWXI	5.999E-23	0.000E+00	0.000E+00

Segredo Perfeito

- Equivocação da chave: H(***K***|***C***) (medida de quanto da chave é revelado pelo criptograma):
  - Para um universo de mensagens ***M*** finito e distribuições de probabilidades de *M* e *K* independentes:
    H(*K*|*C*)= H(*M*|*C*)+H(*K*|(*M*,*C*))
- Segredo perfeito: H(*M*|*C*)=H(*M*)
  - Assim, p(*M*|*C*)=p(*M*)
  - Ou, p(*C*|*M*)=p(*C*)

Segredo Perfeito

- Uma condição necessária a um sistema criptográfico para ter segredo perfeito é que o espaço de chaves seja pelo menos tão grande quanto o espaço de mensagens
  - Um sistema com $|K|=|M|=|C|$ tem segredo perfeito sss toda chave é usada com igual probabilidade, e para todo $M_i \in \boldsymbol{M}$ e todo $C_j \in \boldsymbol{C}$ existe um único $K \in K$, tal que $E_K(M_i)= C_j$
  - Ex: one-time pad

Teorema da Equivocação da Chave

- $H(M,C,K) = H(C) + H(K|C) + H(M|C,K)$
- $H(M,C,K) = H(K) + H(M|K) + H(C|M,K)$

Mas, $H(M|C,K) = H(C|M,K) = 0$ e $H(M|K) = H(M)$

Assim:

$H(K|C)=H(K)+H(M)-H(C)$

- Se $H(K|C) = 0$ => código quebrado.
- Se $H(K|C) > 0$ => duas ou mais chaves podem ter sido utilizadas para produzir o criptograma.
- As possíveis chaves extras são chamadas de espúrias.

Chaves espúrias

- Seja $\boldsymbol{K_C}$ o conjunto de chaves para o qual existem textos em claro que fazem sentido para a cifração $C=E_K(M)$
- O número esperado de chaves espúrias é: $s=\sum_{C\in \boldsymbol{C}} p(C)|\boldsymbol{K_C}|-1$
- $H(K|C)= \sum_{C\in \boldsymbol{C}} p(C)H(K|C)$
- $H(K|C)\leq \log_2|\boldsymbol{K_C}|$
- Usando a desigualdade de Jensen temos
- $H(K|C)\leq \log_2(s+1)$

Distância de Unicidade

- A recuperação da mensagem a partir de um dado criptograma é deve ser função da quantidade de criptograma disponível para análise.
- Para uma mensagem M_n de tamanho n (n-grama), e um criptograma C_n de tamanho n, tem-se:
  - $H(K|C_n) = H(K) + H(M_n) - H(C_n)$;
  - $H(M)\approx nH_L=n(1-R_L)\log_2 m$, onde R_L é a redundância da linguagem e H_L é a entropia da linguagem ($H_{port}\approx 2{,}209$ bits/letra);
  - $H(K) = \log_2|K|$ (chaves são identicamente distribuídas);
  - $H(C)\leq n\log_2 m$;
- Assim:
  - $H(K|C) \geq H(K) - nR_L\log_2 m = \log_2|K| - nR_L\log_2 m$
  - $s \geq |K|/[m^{(nR_L)}]-1$

Distância de Unicidade

- Na média será possível quebrar o sistema criptográfico quando o número de caracteres interceptados é suficiente para fazer o número médio de chaves espúrias igual a zero:
  - $U = \log_2|K|/(R_L \log_2 m)$
  - Exemplo: cifra de substituição $|K|=26!$ e $R_L=75\%$, estime s para n=25 e encontre a distância de unicidade

Dificuldade Computacional

- Algoritmos precisam ser eficientes
- Força bruta: tentativa de todas as chaves possíveis
- Qualquer sistema (exceto segredo perfeito) pode ser quebrado $\Rightarrow$ questão de dinheiro e tempo
- Chaves longas $\Rightarrow$ maior segurança
  - Cifração: O(N+1)
  - Força bruta: $O(2^{N+1})$
- Ferramentas de criptoanálise:
  - Hardware especial
  - Máquinas paralelas
  - ...

Chave secreta x Algoritmo secreto

- Algoritmo secreto ⇒ dificuldade adicional
- Difícil de manter segredo se for largamente utilizado:
  - Engenharia reversa
  - Engenharia social
- Uso comercial: algoritmos publicados
- Uso militar: algoritmo mantido em segredo

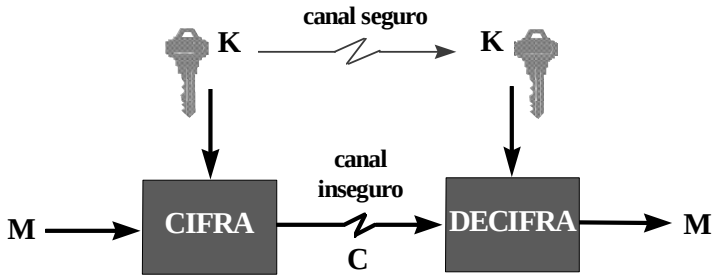
Ataques de Força Bruta

- Número de cifrações/seg: 1 milhão a 1 bilhão bits/sec
- 1999: chave de 56 bits quebrada em 22.5 h com 1800 chips (US$250.000,00) ($245 \times 10^9$ chaves/s, veja eff.org); ajudado por distributed.net
- 1995: chave de 56 bits quebrada em 1 semana com 120.000 processadores (US$6,7M)
- Loteria chinesa:
  - Com máquinas que testam a uma taxa de um milhão de chaves por segundo, necessita-se de 64 segundos para quebrar o DES com um milhão dessas máquinas rodando em paralelo

Segurança Computacional

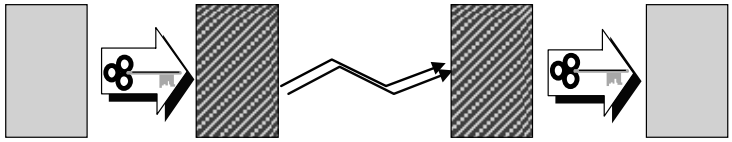
- Difusão: a estrutura estatística herdada do texto em claro é difundida em uma estrutura estatística envolvendo longas combinações de letras no texto.
- Confusão: tem por objetivo fazer a relação entre as estatísticas de C e a descrição do conjunto de chaves muito complexa e intrincada.
  - É implementada usando um conjunto de equações de cifrações não-lineares e múltiplos elementos de chave.

Criptografia Simétrica



Criptografia Simétrica

- Criptografia com ‘Chave Secreta’ ou ‘Simétrica’
- Mesma chave usada para cifrar e decifrar

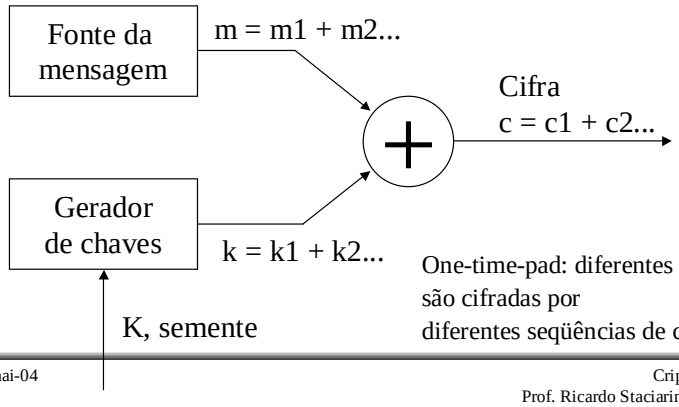


Criptografia Simétrica

- Cifrador seqüencial
  - Cifra uma seqüência de dados digitais um bit ou um byte por vez
- Cifrador de bloco
  - Um bloco do texto em claro é tratado como um todo e usado para produzir um bloco cifrado de tamanho igual

Cifradores Seqüenciais (Stream Ciphers)

- Modelo geral de um cifrador seqüencial



One-time-pad: diferentes mensagens são cifradas por diferentes seqüências de chaves

RC4

- É um cifrador seqüencial com operações orientadas a bytes
- O algoritmo é baseado no uso de uma permutação pseudo-aleatória da chave
- O período do cifrador é aproximadamente 10^{100}
- 8 a 16 operações de máquinas são necessárias por byte (cifrador rápido)
- É usado em protocolos de segurança como o SSL e o WEP

RC4

- Cifrador de cadeia de bytes (byte stream)
- Operação XOR aplicada entre uma seqüência pseudo-randômica chaveada e os dados: a mesma para cifrar e decifrar
- Tamanho da chave variável: até 2048 bits
- Propriedade da RSA, exportável
- Tamanho do código = 1/10 de DES
- Velocidade = de 40 a 100 vezes DES

RC4

- Operações
  - O estado interno do RC4 em um tempo *t* consiste de uma tabela de permutação S_t de 2^n palavras de n bits e de dois ponteiros de n bits i_t e j_t
  - Estas palavras são usadas a cada estado para indexar a palavras diferentes tabela S_t, realizando sempre adições módulo 2^n

RC4

- Ataques ao RC4
  - Chave de 40 bits é fraca
  - Ataque de VI conhecido e escolhido
  - Ataque de invariância

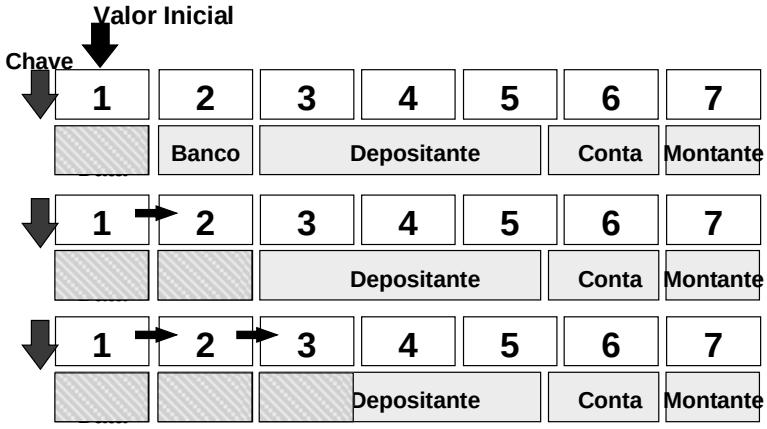
A5

- A5/1 e A5/2 são cifradores seqüenciais para sigilo de conversações de telefones GSM
- Chaves de 64 bits geram seqüências de chaves de 228 bits cada para cifração de um quadro
  - Utiliza três registradores de deslocamento e uma transformação não-linear para gerar a seqüência de chave

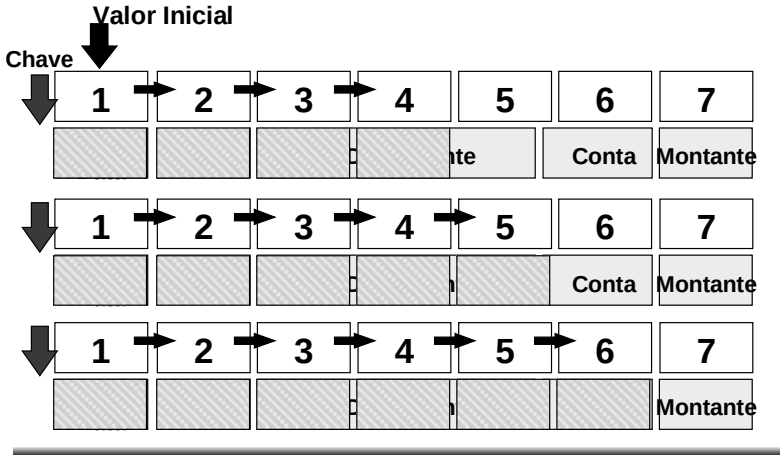
Cifradores de bloco

- Blocos e chaves de tamanho fixo
  - Não podem ser muito pequenos ⇒ segurança comprometida
  - Não devem ser muito grandes ⇒ problemas de desempenho
  - Tamanho típico de bloco: 64 bits
- Mensagem deve ser quebrada em blocos
- Texto em claro e texto cifrado não devem ter correlação
  - metade deve permanecer igual e metade diferente
  - espalhamento de bits

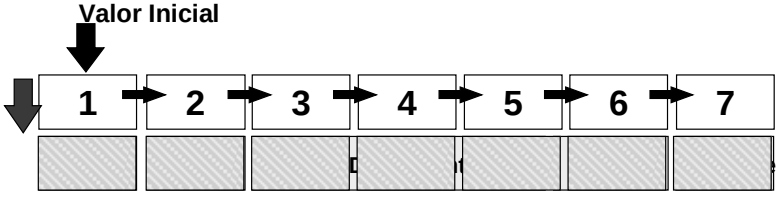
Cifradores de Bloco



Cifradores de Bloco



Cifradores de Bloco

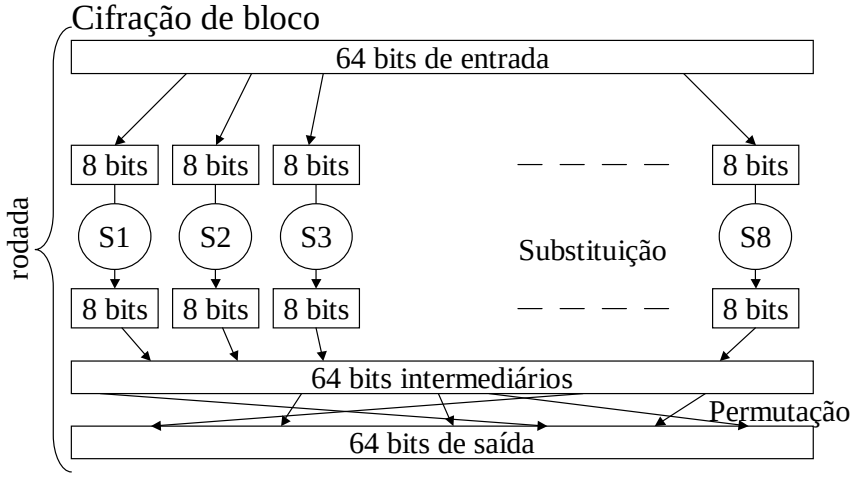


- Cifragem em blocos, usualmente de 64-bit;
- Cifragem de blocos em cadeia (CBC) bastante comum;
- Em geral, mais lenta que a cifragem em fluxo.

Cifradores de Bloco

- Conceitos gerais:
  - Substituição (alfabeto permutado): mapeamento de um bloco de k bits em 2^k blocos de k bits
  - Transposição (permutação no bloco): mudança da posição de cada bit no bloco
  - Round (rodada): combinação de substituição de grupos de bits com permutação, de tal forma que um bit da entrada possa afetar todos os bits da saída

Cifração de bloco



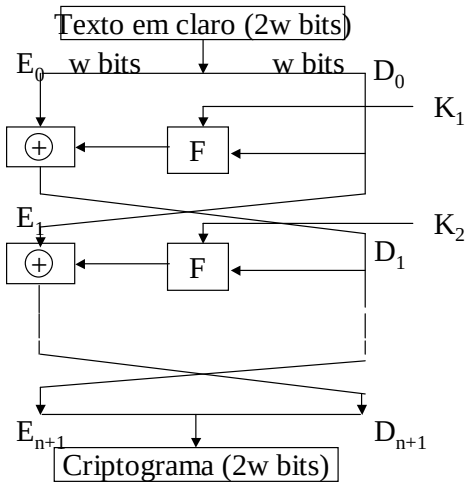
Rede Feistel

- Objetivos:
  - Difusão
    - Refere-se ao remanejamento dos bits na mensagem tal que qualquer redundância do texto em claro é espalhada sobre o criptograma
    - Uma transposição em uma rodada é dita adicionar difusão
  - Confusão
    - Visa tornar o relacionamento entre a chave e o criptograma tão complexo quanto possível
    - Uma substituição em uma rodada é dita adicionar confusão

Rede Feistel

- Parâmetros:
  - tamanho do bloco
  - tamanho da chave
  - número de rodadas
  - algoritmo de geração de subchaves
  - função de rodada

Rede Feistel



DES

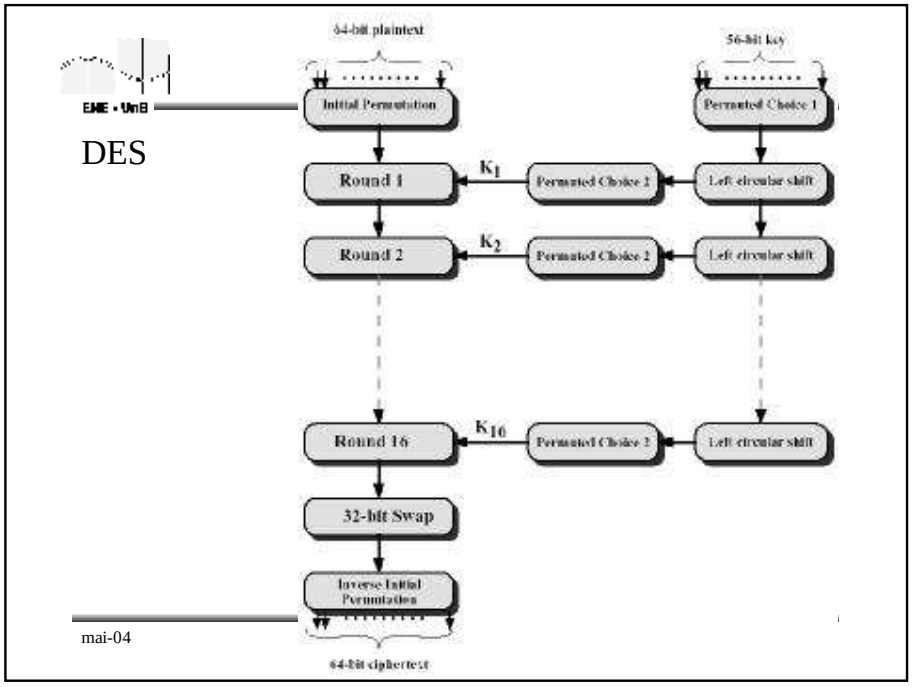
- Publicado em 1977 pelo National Bureau of Standards
- Desenvolvido pela IBM (Lucifer)
- Chaves de 56 bits com paridade
- Blocos de 64 bits
- Fácil de implementar em hardware, lento em software (na época)
- Throughput superior a 20 Mbps em máquinas típicas

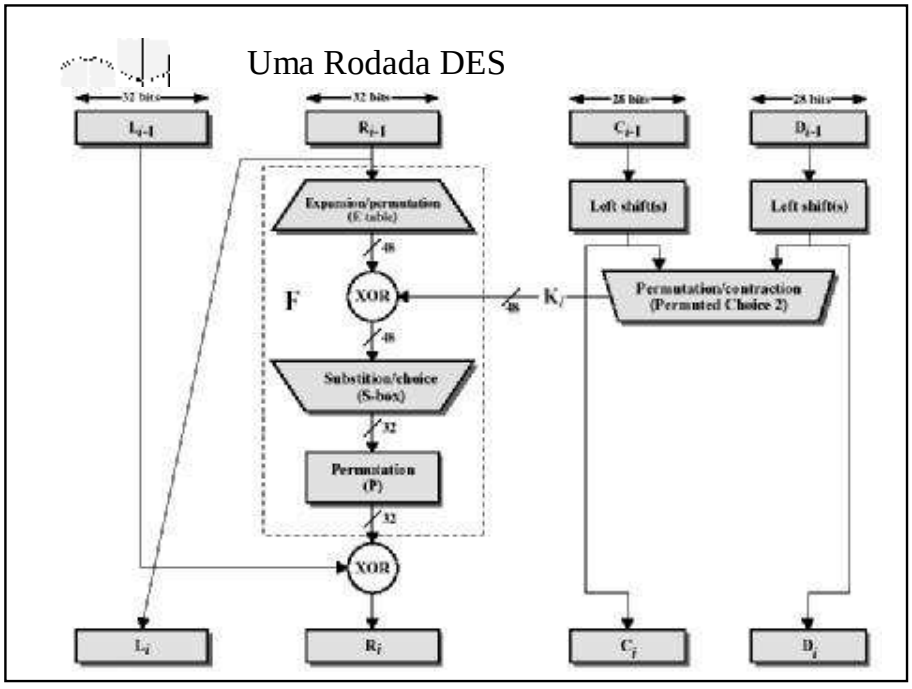
Visão Geral do DES

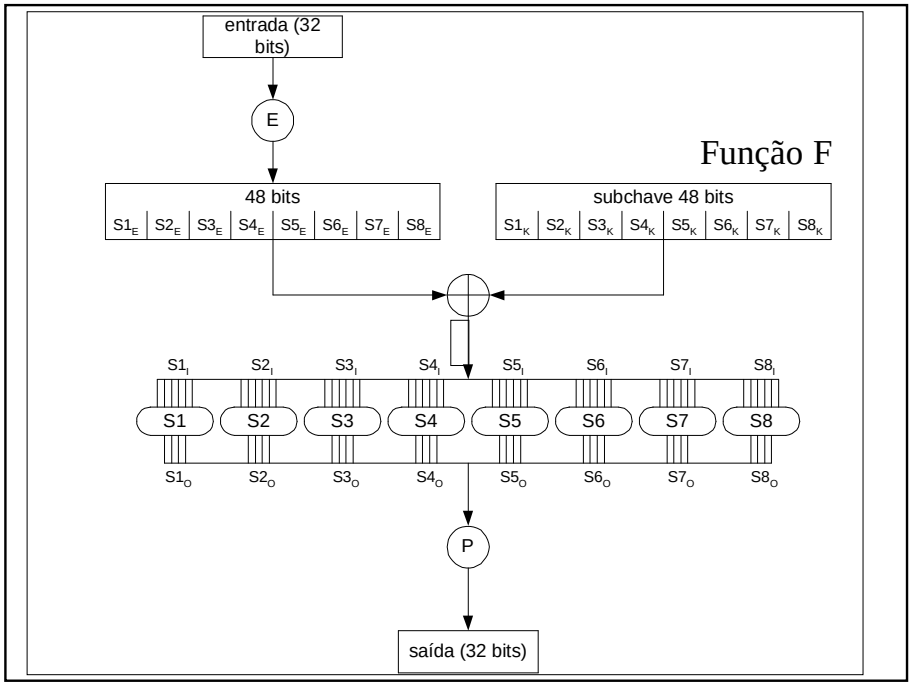
- Permutação inicial
- Chave de 56 bits ⇒ 16 chaves por round de 48 bits
- 16 rodadas: 64 bits de entrada + 48 bits de chave → 64 bits de saída
- Permutação final (inversa da inicial)
- Decifração: execução ao contrário com a chave na ordem inversa

DES: geração das chaves de rodadas

- Bits 8, 16, ..., 64 são de paridade
- Permutação inicial dos 56 bits válidos
- Divisão em dois grupos de 28 bits
- A cada rodada a chave é deslocada um ou dois bits à esquerda
- Permutação entre as metades esquerda e direita da chave
- Descarte de 8 bits por rodada


64-bit plaintext
56-bit key
Initial Permutation
Permuted Choice 1
DES
Round 1
K_1
Permuted Choice 2
Left circular shift
Round 2
K_2
Permuted Choice 2
Left circular shift
Round 16
K_{16}
Permuted Choice 2
Left circular shift
32-bit Swap
Inverse Initial Permutation
mai-04
64-bit ciphertext


Uma Rodada DES
32 bits
32 bits
28 bits
28 bits
L_{i-1}
R_{i-1}
C_{i-1}
D_{i-1}
Expansion/permutation
(E-table)
Left shift(s)
Left shift(s)
48
F
XOR
K_i
48
Permutation/contraction
(Permuted Choice 2)
48
Substition/choice
(S-box)
32
Permutation
(P)
32
XOR
L_i
R_i
C_i
D_i



Ataques ao DES

- Força bruta: 0,5 MMY
- É necessário saber alguma informação sobre o texto em claro (ASCII, GIF, ...)
- Quebras documentadas usando hardware dedicado (custom chip ou lógica programável)
  - www.eff.org
  - distributed.net
- Pode-se usar cifrações repetidas para melhorar segurança

Ataques ao DES

- criptoanálise diferencial
  - Biham e Shamir, 92
  - ataque de mensagem escolhida
  - baseada em propagação de diferenças conhecidas nas mensagems em claro
  - no caso do DES: são necessários 2^{47} pares de mensagens em claro e criptogramas

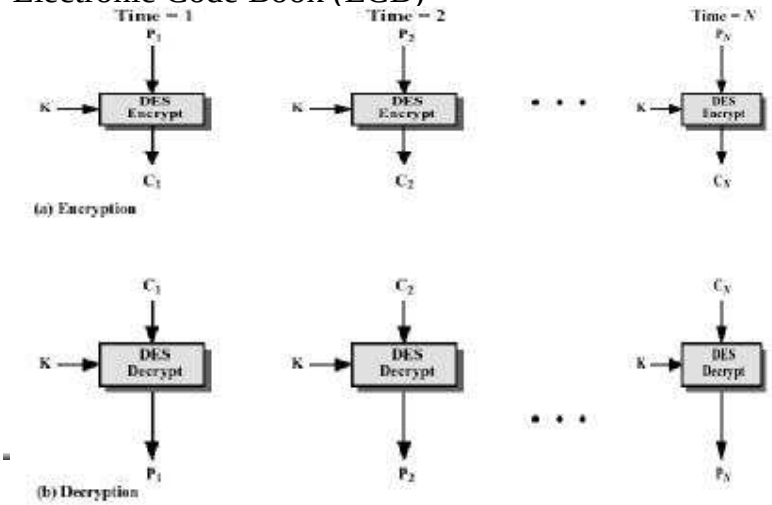
Ataques ao DES

- criptoanálise linear
  - Matsui, 93
  - ataque de mensagem conhecida
  - procura encontrar expressões lineares para as caixas-S, estendendo estas expressões para a cifra inteira.
  - no caso do DES: são necessários 2^{47} pares de mensagens em claro e criptogramas

Modos de Operação do DES

- Electronic Code Book (ECB)
  - Cada bloco de 64 bits de texto aberto é codificado independentemente, usando a mesma chave
  - Transmissão segura de valores únicos (por exemplo, uma chave de criptografia)

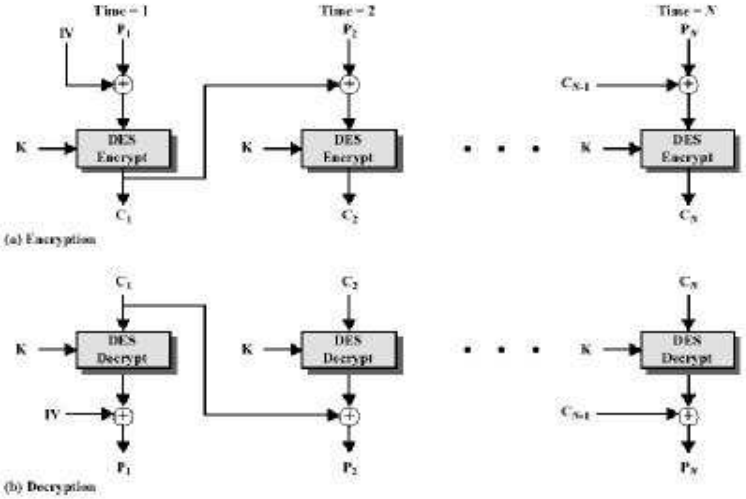
Electronic Code Book (ECB)



Modos de Operação do DES

- Cipher Block Chaining (CBC)
  - A entrada para o algoritmo de criptografia é o resultado da operação XOR do próximo bloco de 64 bits de texto aberto e o bloco precedente de 64 bits de texto cifrado
  - Transmissão orientada a blocos de propósito geral
  - Autenticação

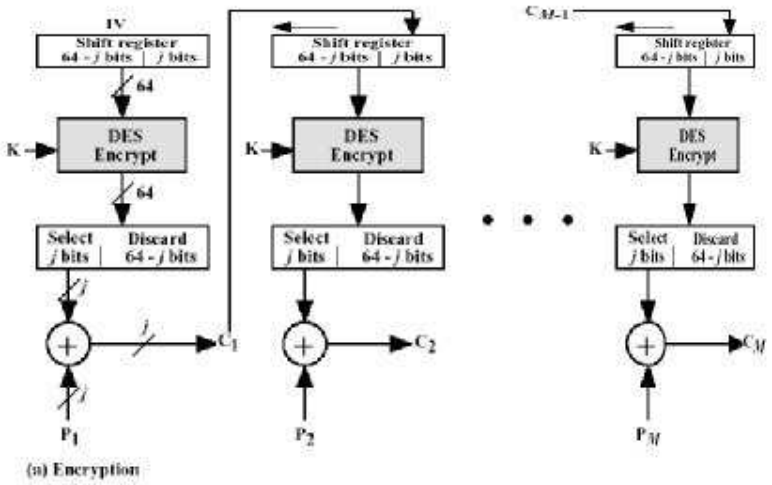
Cipher Block Chaining (CBC)



Modos de Operação do DES

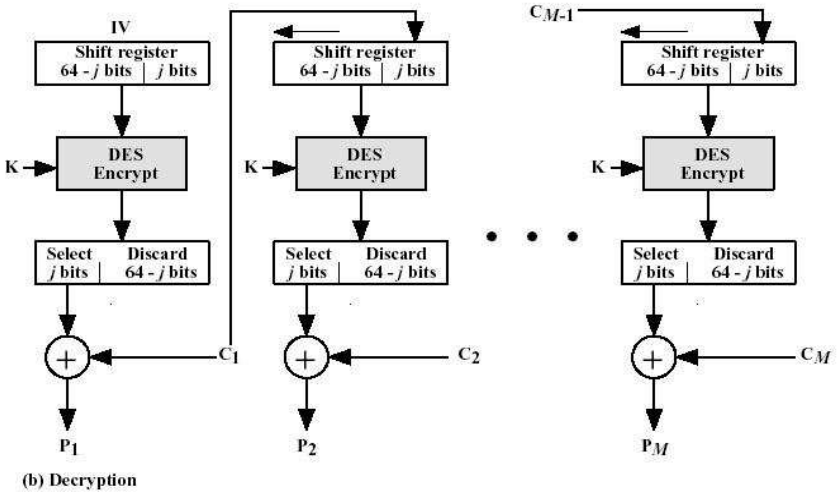
- Cipher Feedback (CFB)
  - Processa-se **j** bits de cada vez. O texto cifrado precedente é usado como entrada para o algoritmo de criptografia para produzir uma saída pseudo-aleatória, que é, então, operada usando XOR com o texto em claro para produzir a próxima unidade de texto cifrado.
  - Transmissão orientada a cadeias de propósito geral

Cipher Feedback (CFB)



(a) Encryption

Cipher Feedback (CFB)

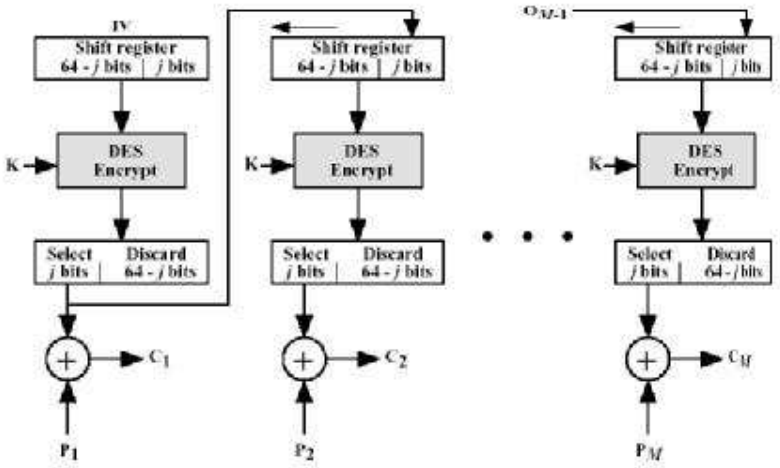


(b) Decryption

Modos de Operação do DES

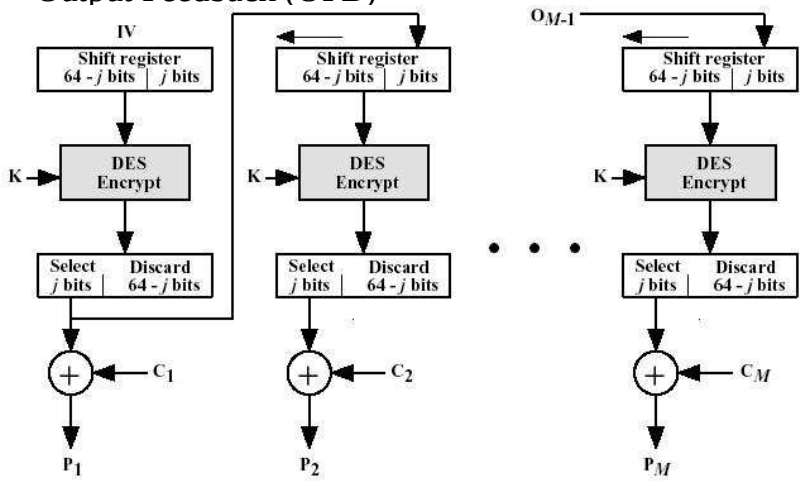
- Output Feedback (OFB)
  - Similar ao CFB, exceto que a entrada para o algoritmo de criptografia é a saída DES precedente
  - Transmissão orientada a cadeias sob canal ruidoso (por exemplo, comunicação por satélite)

Output Feedback (OFB)



(a) Encryption

Output Feedback (OFB)



(b) Decryption

Geração de MICs

- Message Integrity Code
  - Enviar somente o último bloco do CBC (resíduo CBC)
  - Qualquer modificação no texto em claro modifica o resíduo CBC

3-DES

- Cifração-decifração-cifração (EDE):
  - Duas chaves: K_1, K_2

$$M \rightarrow \underset{K_1}{E} \longrightarrow \underset{K_2}{D} \longrightarrow \underset{K_1}{E} \longrightarrow C$$

  - Decifração

$$C \rightarrow \underset{K_1}{D} \longrightarrow \underset{K_2}{E} \longrightarrow \underset{K_1}{D} \longrightarrow M$$

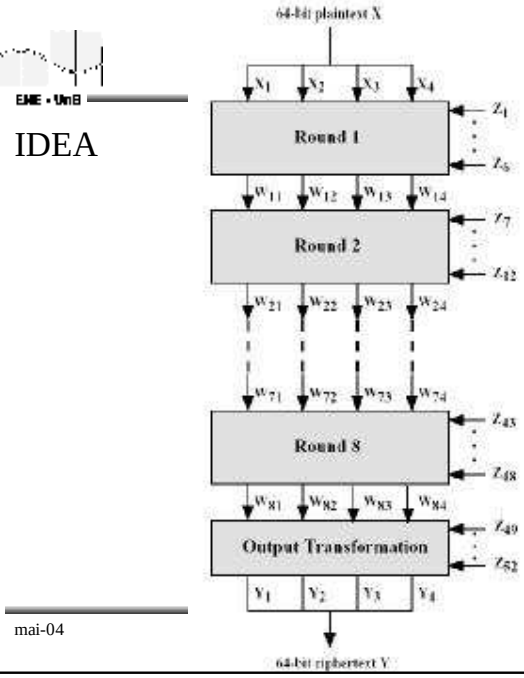
3-DES

- Ataque
  - encontro-no-meio (meet-in-the-middle)
    - Decifra-se o resultado da segunda cifra e armazena-se todas as possíveis entradas
    - Cifra-se todos os textos em claro com a primeira cifra e compara-se com a tabela
    - As coincidências são armazenadas como possíveis pares corretos de chaves

IDEA

- International Data Encryption Algorithm (originalmente IPES)
- Autores: Lai e Massey da ETH Zurich, 1991
- Blocos de 64 bits
- Chaves de 128 bits

IDEA



IDEA

- Operações primitivas
  - $\oplus$
  - + mod 2^{16}
  - $\otimes$ mod $2^{16}+1$:
    - Inversível: $x \otimes y = 1 \bmod 2^{16}+1$
    - Razão: $2^{16}+1$ é primo
    - Trata 0 como código para 2^{16}

IDEA

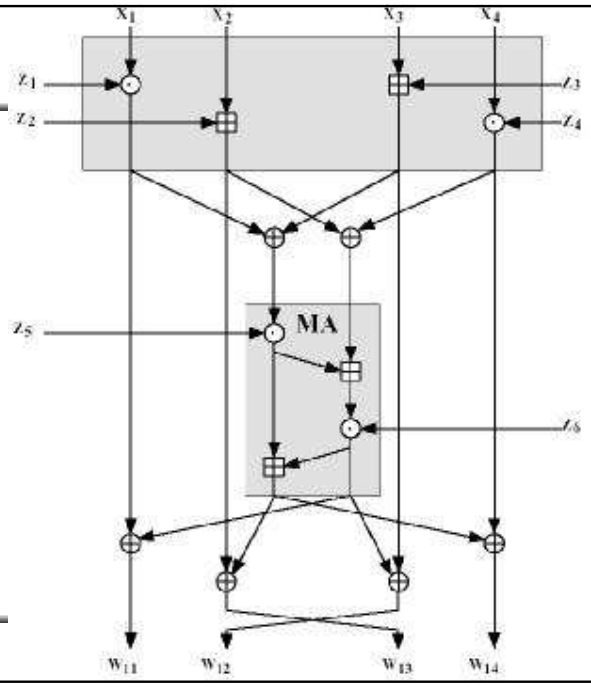
- Expansão da chave:
  - Chave de 128 bits → 52 chaves de 16 bits
  - Cifração e decifração usam chaves diferentes
  - Geração da chave
    - Divide-se os 128 bits em o subchaves de 16 bits
    - Começando no bit 25, toma-se mais 8 chaves de 16 bits
    - Deslocamento de 25 bits e repetição da operação

IDEA

- 17 rodadas:
  - Pares e ímpares
  - 64 bits → 4 entradas de 16 bits: X_a, X_b, X_c, X_d
  - Saídas: X_a', X_b', X_c', X_d'
  - Rounds ímpares usam quatros chaves de entrada: K_a, K_b, K_c, K_d
  - Rounds pares usam somente 2 chaves: K_e, K_f

IDEA

Uma rodada



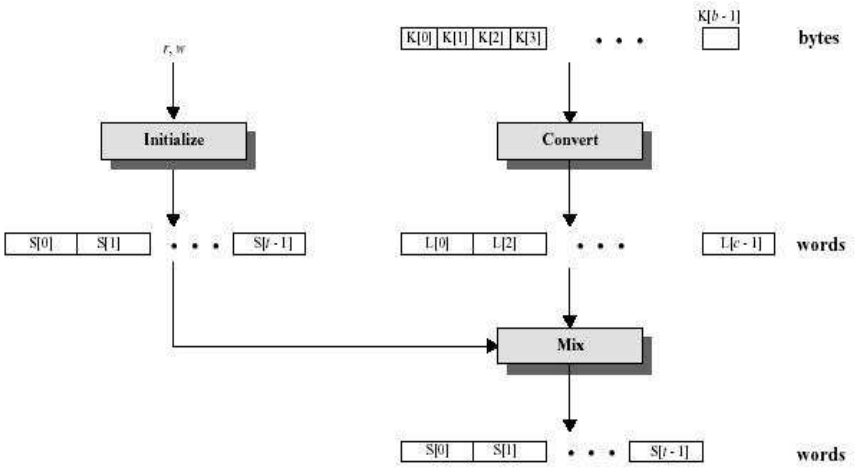
RC5

- Características:
  - Tamanho de palavra variável (w)
    - valores permitidos: 16, 32, 64
  - Número variável de rodadas (r)
    - valores permitidos: 0, 1, …, 255
  - Chave de tamanho variável (b)
    - número de bytes permitidos: 0, 1, …, 255
  - Exemplo: RC5-32/12/16

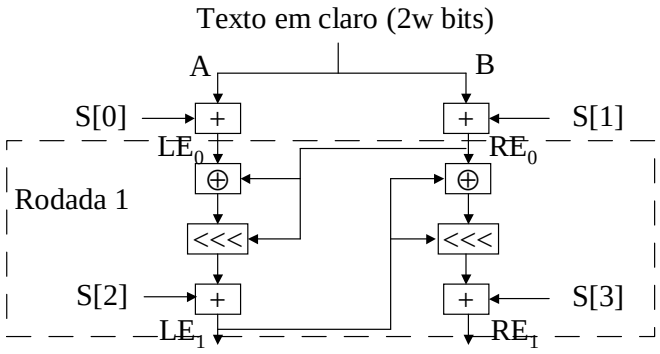
RC5

- Operações
  - Expansão de chave: t=2r+2
    - S[0], S[1], …, S[t-1]
  - Cifração
    - adição módulo 2^w
    - ou exclusivo
    - rotação circular à esquerda

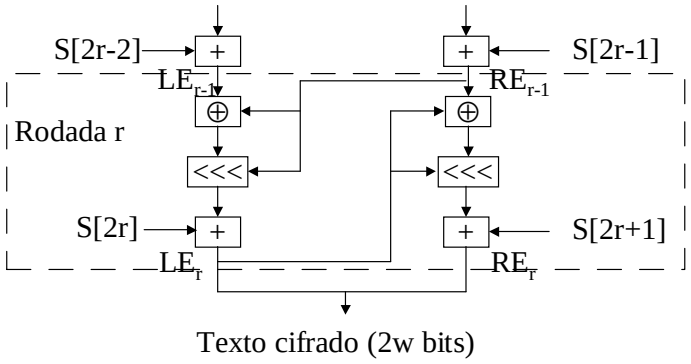
RC5 - Expansão da Chave



RC5


Texto em claro (2w bits)
A
B
S[0]
S[1]
LE0
RE0
Rodada 1
<<<
<<<
S[2]
S[3]
LE1
RE1

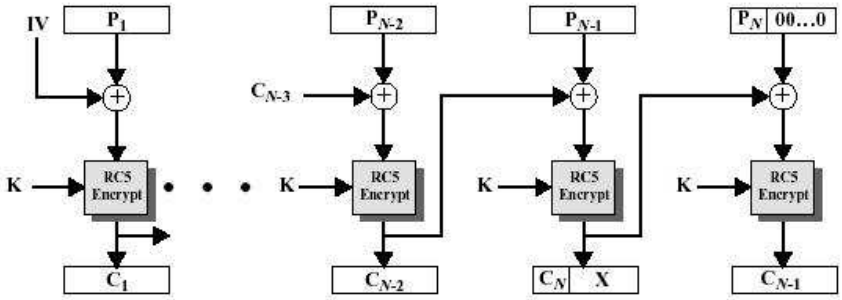
RC5


S[2r-2]
S[2r-1]
LEr-1
REr-1
Rodada r
<<<
<<<
S[2r]
S[2r+1]
LEr
REr
Texto cifrado (2w bits)

RC5

- Modos de operação
  - RC5 block cipher
    - semelhante ao ECB
  - RC5-CBC
  - RC5-CBC-Pad
  - RC5-CTS (ciphertext stealing mode)

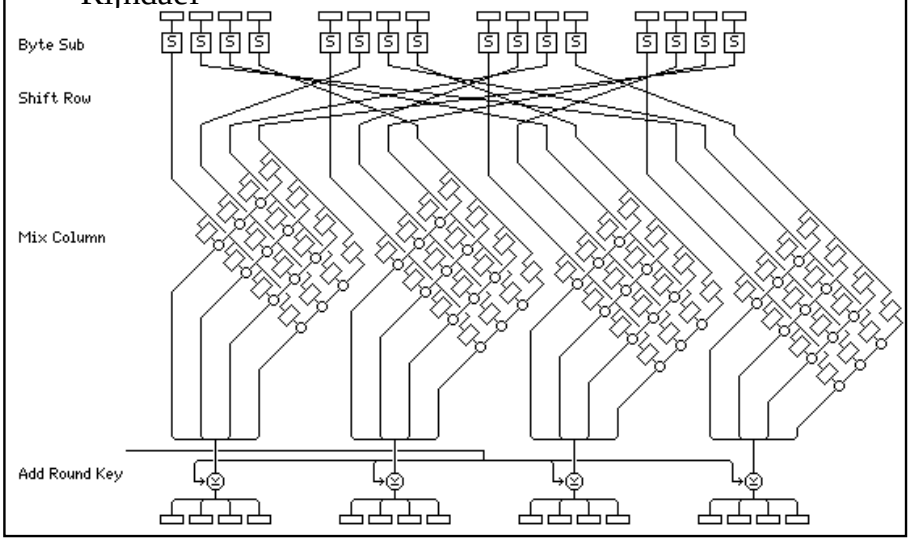
RC5 - stealing mode



Rijndael

- Características
  - operações realizadas no nível de byte
  - tamanhos de chaves: 128, 192, 256 bits
  - tamanhos de blocos: 128, 192, 256 bits
  - número de rodadas: 9, 11, 13 (além de uma rodada extra no final)
  - cada rodada composta de quatro passos: substituição de bytes, deslocamento de linhas, mistura de colunas, soma da sub-chave de rodada

Rijndael



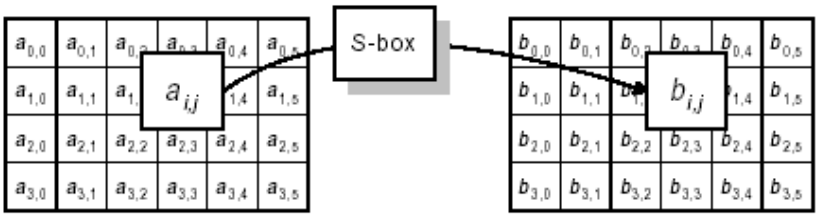
Rijndael - Substituição de Bytes

- Transformação definida por

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Rijndael - Substituição de Bytes

- Efeito da substituição de bytes



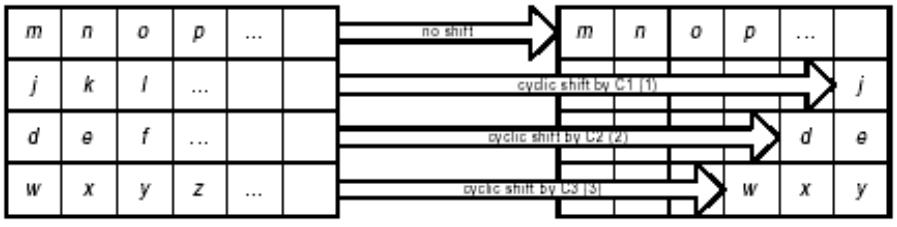
Rijndael – Deslocamento de Linhas

- Deslocamentos definidos pela tabela abaixo, conforme o tamanho do bloco Nb

Nb	C1	C2	C3
4	1	2	3
6	1	2	3
8	1	3	4

Rijndael – Deslocamento de Linhas

- Efeito do deslocamento de linhas



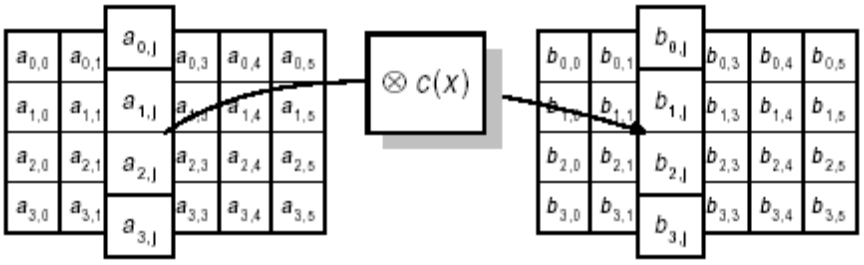
Rijndael – Mistura de Colunas

- Transformação sobre as colunas

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

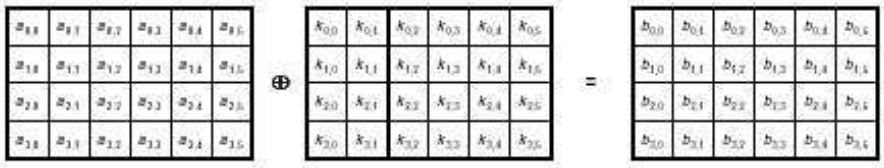
Rijndael – Mistura de Colunas

- Efeito da mistura de colunas



Rijndael – Soma de Sub-Chave de Rodada

- Operação XOR



Rijndael

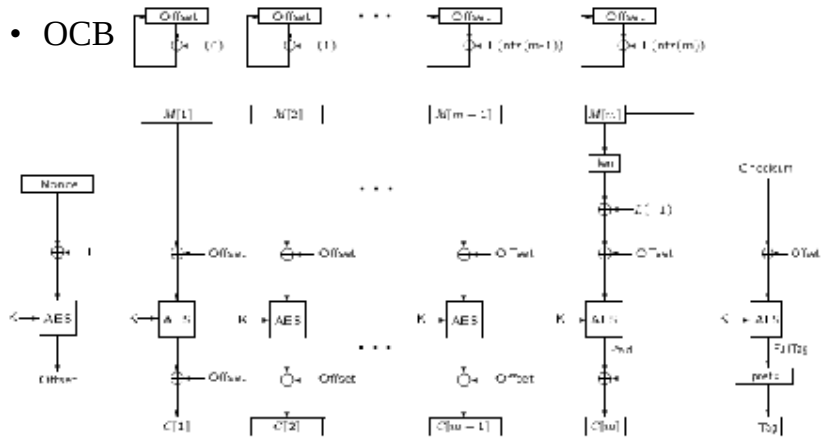
- última rodada
  - igual às anteriores sem a mistura de colunas
- geração das sub-chaves de rodada
  - chave original expandida e seleção da chave de rodada
  - expansão baseada
    - em rotação
    - caixas-S da substituição de bytes
    - XOR com constante da rodada

Rijndael

- Modos de operação
  - CTR (Counter mode): modificação do OFB
  - OCB (Offset Codebook): permite autenticação
- Ataques
  - Pouco susceptível às criptoanálises diferencial e linear
  - Há dúvidas sobre sua resistência a ataques algébricos

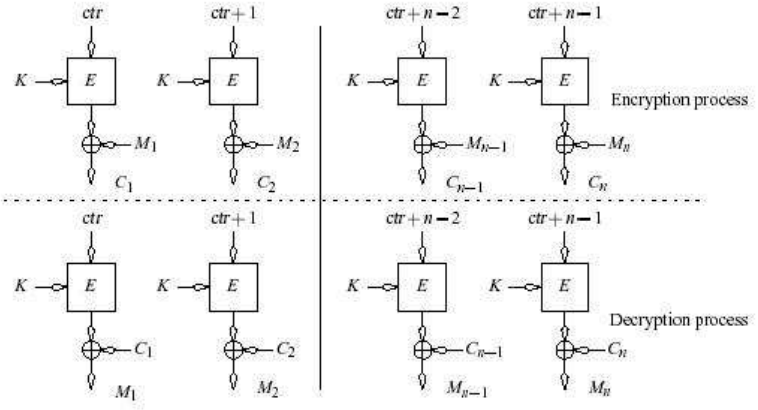
Rijndael

- OCB



Rijndael

- Counter Mode



Criptografia Simétrica – Vantagens e Desvantagens

- Vantagens
  - Relativamente Segura
  - Amplamente utilizada
  - Rápida
- Desvantagens
  - Requer compartilhamento da chave secreta
  - Chaves **devem** ser guardadas com segurança
  - Grande número de chaves
  - Administração complexa
  - Não permite a não-retratação

Funções de Autenticação

- Três classes de autenticadores
  - cifração de mensagens
    - o criptograma da mensagem inteira serve como autenticador
  - código de autenticação de mensagem (MAC)
    - função da mensagem e de uma chave secreta que produz um valor de comprimento fixo
  - funções de condensação
    - função que mapeia uma mensagem de qualquer comprimento em um valor resumo de tamanho fixo

Funções *Hash*

- Funções hash ou de condensação são funções unidirecionais: $d = h(m)$, mas não é tratável $m = h^{-1}(d)$.
  - Levam mensagens de tamanho arbitrário em resumos (hash) de tamanho fixo
- Funções hash tem como premissa:
  - evitar que se encontre a mensagem dado o resumo
  - evitar que se encontre m_1, m_2 tal que $d_1=d_2$
  - dado m_1, evitar que se encontre m_2 com mesmo resumo

Funções *Hash*

- Aleatoriedade:
  - Distribuição uniforme
    - para muitas entradas, as saídas devem ser ‘1’ na metade do tempo
    - cada saída deve ter 50% de bits ‘1’
  - Independência
    - entradas parecidas devem levar a saídas não relacionadas (metade dos bits deve diferir na saída)

Funções *Hash*

- Muitas mensagens podem levar ao mesmo resumo:
  - Suponha resumos de 128 bits e mensagens com 1000 bits de comprimento
  - Na média, existem 2^{872} mensagens que mapeiam para o mesmo resumo
  - Mas, na média, são necessárias 2^{128} tentativas para se encontrar duas tais mensagens

Funções *Hash*

- Paradoxo do aniversário:
  - quantas pessoas deve haver numa sala para que se tenha 50% de chances de se encontrar duas pessoas que possuam a mesma data de aniversário?
  - Seja *n* a quantidade de entradas
  - Sejam *k* = 365 as possíveis saídas
  - Há no total n(n-1)/2 pares de entrada
  - Probabilidade de não haver colisão: $p=(1/k)^{n(n-1)/2}$
  - Por exemplo, se n=23, $p_{todosdiferentes}=0,49$

Funções *Hash*

- Paradoxo do Aniversário
  - Para m bits, são necessários cerca de $2^{m/2}$ mensagens para que se encontre uma colisão
  - Lembrando que este problema é diferente de pegar h e encontrar o m relativo
  - Exemplo:
    - Alex, cuja secretária é Bia quer demitir Melo
    - Bia é amiga de Melo
    - Bia compõe duas mensagens com o mesmo hash: uma para ser lida por Alex e outra para ser enviada para Melo

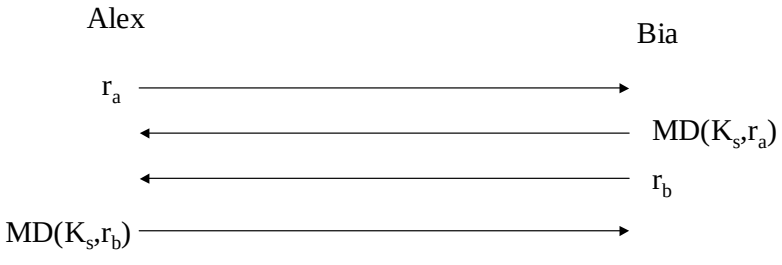
Funções *Hash*

- Exemplos:
  - RFC: www.normos.org, www.rfc-editor.org

Nome	Inventor	Documentação	Objeto	Hash
MD2	Ron Rivest	RFC1319	bytes	128
MD4	Ron Rivest	RFC1320	32-bits	128
MD5	Ron Rivest	RFC1321	32-bits	128
SHA-1	NIST	FIPS PUB 180	32-bits	160

Funções *Hash*

- Autenticação usando MAC=MD[$K_s|m_1$]



Funções *Hash*

- Cifração: one-time pad
  - b_1=MD[K_s|IV]
  - b_2=MD[K_s|b_1]
  - b_i=MD[K_s|b_{i-1}]
  - b é usado como one-time pad

Funções *Hash*

- Cifração: misturando no texto em claro
  - similar ao CFB
  - quebra a mensagem em grupos de comprimento do resumo p_i
  - b_1=MD[K_s|IV], c_1=p_1⊕b_1
  - b_2=MD[K_s|c_1], c_2=p_2⊕b_2
  - b_i=MD[K_s|c_{i-1}], c_i=p_i⊕b_i

MD2

- Características
  - resumo de 128 bits
  - pad para múltiplo de 16 bytes
  - anexa MD2 checksum no final
  - processa mensagens inteiras

MD2

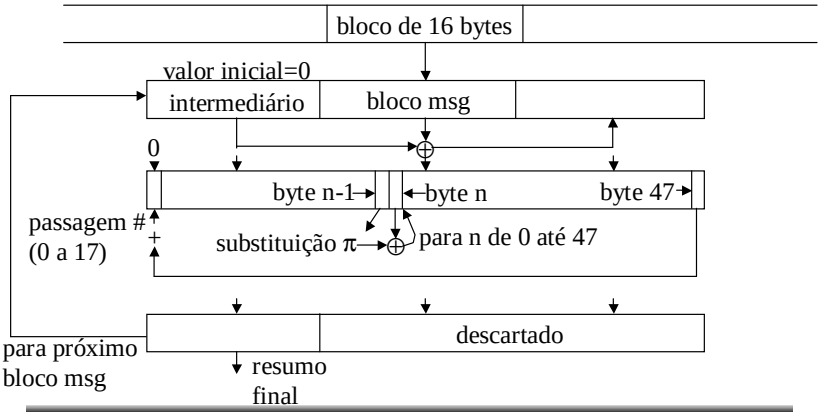
- Checksum MD2
  - m_{nk}: byte *nk* da mensagem
  - $c_n=\pi(m_{nk} \oplus c_{n-1})$
  - π: tabela de substituição
    - $0 \rightarrow 41$
    - $1 \rightarrow 46$
    - ...

MD2

- Passagem final
  - opera em grupos de 16 bytes
  - quantidade q de 48 bytes:
    - (resumo corrente|grupo|resumo ⊕grupo)
  - 18 passagens sobre q
  - $c_n = c_n \oplus \pi(c_{n-1})$ para n=1...47, $c_{-1}=0$
  - após 17 passagens usa os primeiros 16 bytes como o novo resumo

MD2

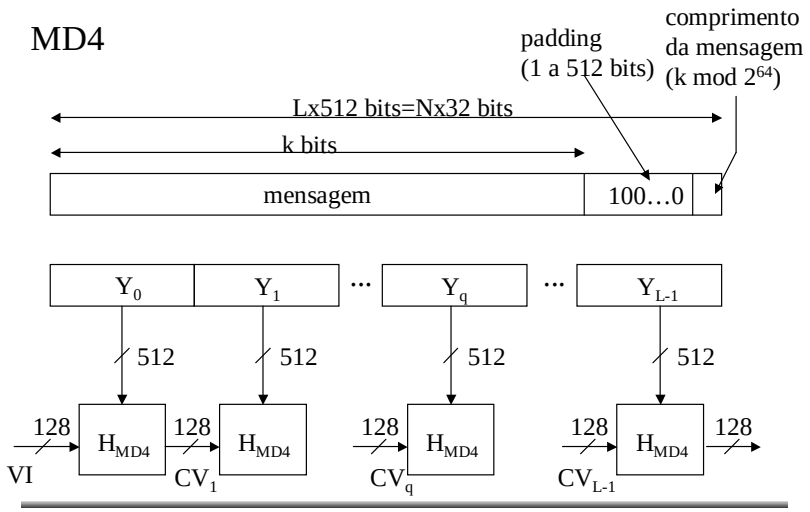
- Passagem final



MD4

- Características:
  - qualquer número de bits
  - opera sobre quantidades de 32 bits
  - faz pad para múltiplos de 512 bits
  - resumo=f(blocos de 512 bits, resumo anterior)

MD4



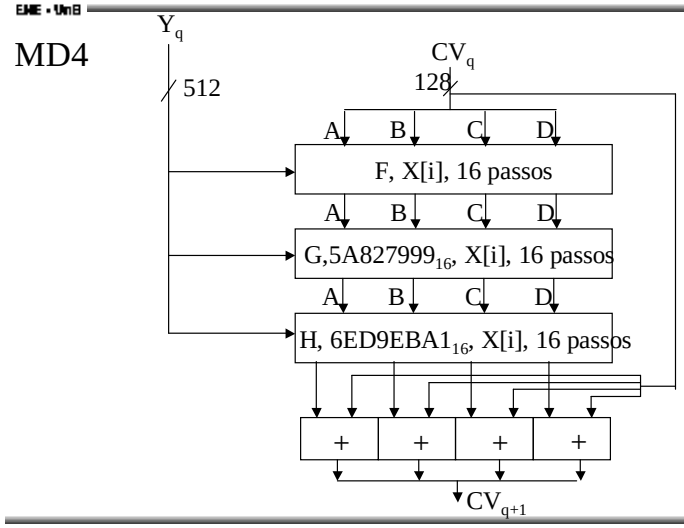
MD4

- Operações
  - $\lfloor x \rfloor$
  - ~: complemento de bit
  - $x \wedge y$
  - $x \vee y$
  - adição módulo 2^{32}
  - rotação à esquerda por y bits

MD4

- Passagem 1: seleção
  - função de seleção: $F(x,y,z)=(x \wedge y) \vee (\sim x \wedge z)$
- Passagem 2: majoração
  - função de majoração: $G(x,y,z)=(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$
- Passagem 3: xor
  - função: $H(x,y,z)=x \oplus y \oplus z$
- Obs
  - A cada um dos 16 passos de cada passagem, é feita uma rotação à direita com A, B, C, D (descritos no esquema seguinte), passando as 3 últimas palavras a assumir os papéis de x, y e z

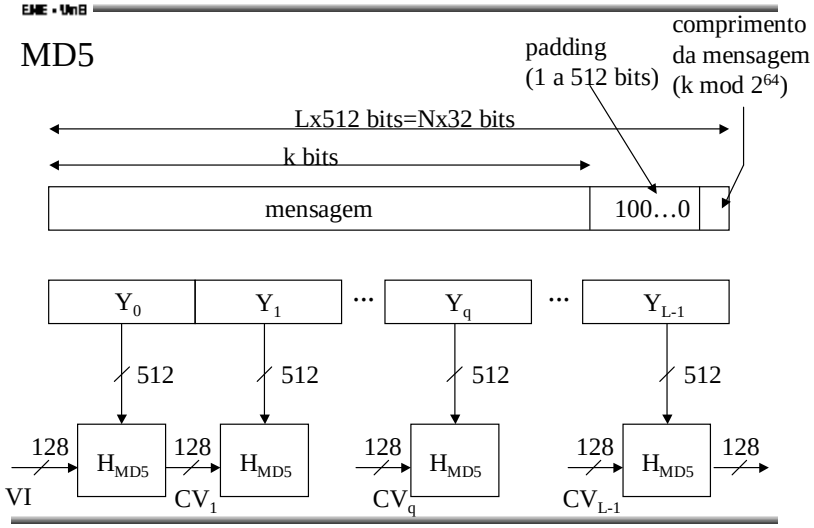
MD4



MD5

- Características:
  - mesmo padding do MD4
  - MD5: 4 passagens

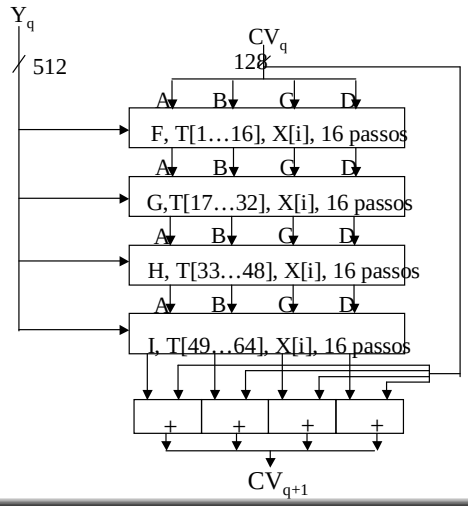
MD5



MD5

- Passagem 1 e 3:
  - como em MD4
- Passagem 2:
  - função $G(x,y,z)=(x \wedge z) \vee (y \wedge \sim z)$
- Passagem 4:
  - $I(x,y,z)=y \oplus (x \vee \sim z)$
- Obs
  - A cada um dos 16 passos de cada passagem, é feita uma rotação à direita com A, B, C, D (descritos no esquema seguinte), passando as 3 últimas palavras a assumir os papéis de x, y e z

MD5



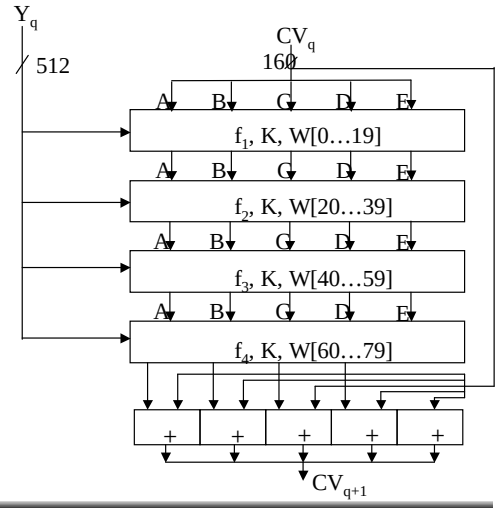
SHA-1

- Características:
  - 64 bits para 160 bits
  - similar ao MD4
  - 5 passagens
  - Resumo:
    - CV_0=IV
    - $CV_{q+1}=SOMA_{32}(CV_q,ABCDE_q)$
    - $MD=CV_L$
    - onde
      - $ABCDE_q$= saída do último round de processamento do q-ésimo bloco
      - L=número de blocos da mensagem
      - $SOMA_{32}$= soma módulo 32

SHA-1

- Passagem 1: seleção
  - função $f_1(x,y,z)=(x \wedge y) \vee (\sim x \wedge z)$
- Passagem 2:
  - função $f_2(x,y,z)= x \oplus y \oplus z (x \wedge z) \vee (y \wedge \sim z)$
- Passagem 3: Xor
  - função $f_3(x,y,z)=(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$
- Passagem 4:
  - função $f_4(x,y,z)= x \oplus y \oplus z$

SHA-1



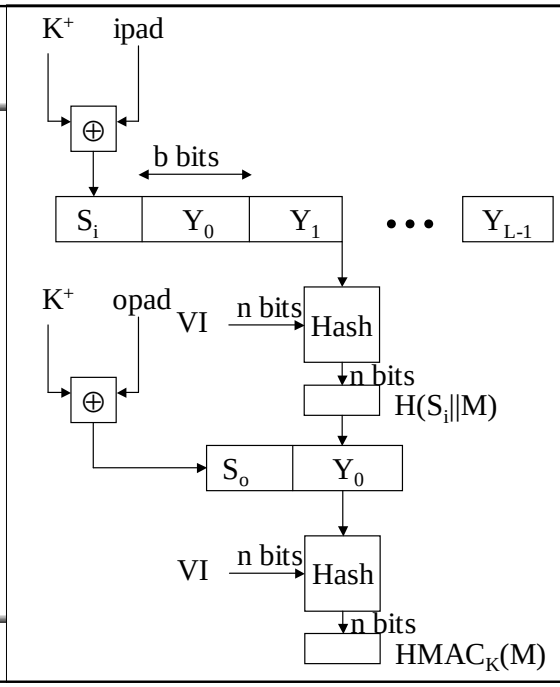
HMAC

- Objetivos de projeto RFC 2104
  - usar, sem modificações, funções de condensação disponíveis
  - permitir fácil substituição da fção condens.
  - preservar desempenho original da fção condens.
  - usar e manipular chaves de forma simples
  - permitir análise criptográfica de sua força

HMAC

- Operação
  - $HMAC_K = H[(K^+ \oplus opad) || H[(K^+ \oplus ipad) || M]]$
    - H: fção condens.
    - M: msg
    - Y_i: i-ésimo bloco de M
    - L: nr de blocos em M
    - b: número de bits num bloco
    - n: comprimento do resumo
    - K: chave secreta
    - K^+: K preenchida com zeros à esq até b bits de compr.
    - ipad: 00110110 repetido b/8 vezes
    - opad: 01011010 repetido b/8 vezes

HMAC



Criptografia Assimétrica

- Problemas da criptografia convencional
  - na distribuição de chaves os comunicantes devem
    - já compartilhar a chave
    - ter um centro de distribuição de chaves
  - não possibilita assinatura digital

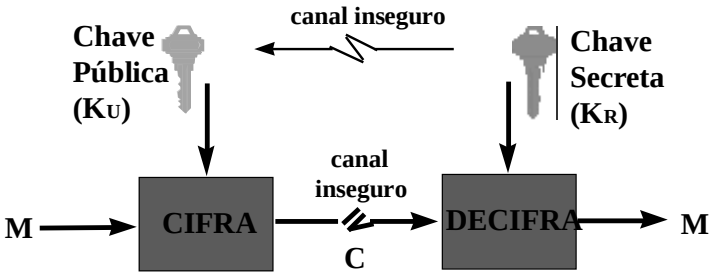
Criptografia Assimétrica

- Algoritmos de chave pública baseiam-se em uma chave para cifração e outra para decifração
  - Deve ser computacionalmente intratável determinar a chave de decifração dado somente conhecer o algoritmo de criptografia e a chave de cifração
  - Em alguns algoritmos as chaves relacionadas podem ser usadas para cifração, com a outra sendo usada para decifração

Criptografia Assimétrica

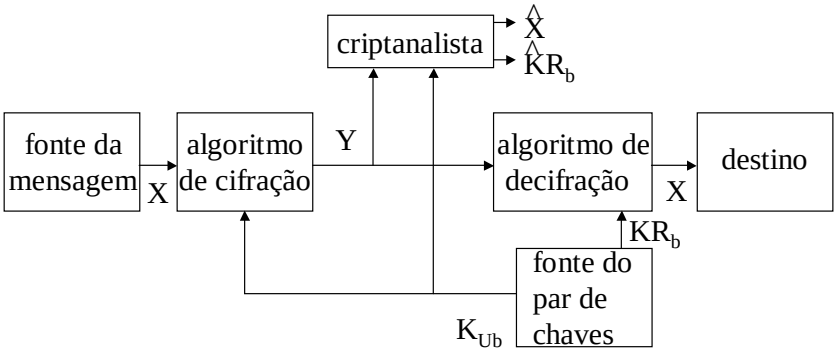
- Os passos essenciais de um processo de cifração de chave pública
  - Cada sistema final numa rede gera um par de chaves
  - Cada sistema publica sua chave de cifração colocando-a em um registro ou arquivo público
  - Se Alex deseja enviar uma mensagem para Bia, ele envia a mensagem usando a chave pública de Bia
  - Quando Bia recebe a mensagem, ela a decifra usando sua chave privada

Criptografia Assimétrica



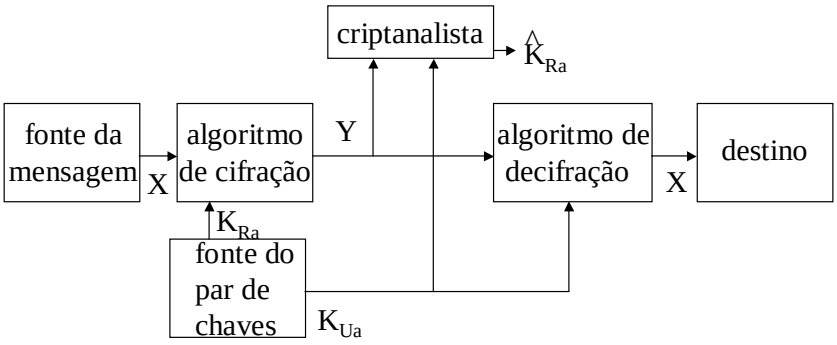
Criptografia Assimétrica

– Confidencialidade



Criptografia Assimétrica

– Autenticação



Criptografia Assimétrica

- Categorias de algoritmos de chave pública:
  - Cifração/decifração
    - Ex: RSA
  - Assinatura digital
    - Ex: RSA, DSS
  - Troca de chaves
    - Ex: RSA, Diffie-Hellman

Criptografia Assimétrica

- Condições dos algoritmos de chave pública
  - fácil de gerar o par K_U e K_R
  - fácil de computar $C=E_{KU}(M)$
  - fácil de computar $M=D_{KR}(C)$
  - difícil de determinar K_R a partir de K_U
  - difícil de computar M, dado K_U e C
  - $M=E_{KU}[D_{KR}(M)]$

Básico de Teoria dos Números

- Divisibilidade
  - símbolo "|" significa divide
    - a|b: a divide b ou b é divisível por a
  - propriedades
    - se a|1, então a=±1
    - se a|b e b|a, então a=±b
    - todo b≠0 divide 0
    - se b|g e b|h, então b|(mg+nh) para m e n inteiros arbitrários
    - ex: 7|(3x14+2x63), pois 7\7(3x2+2x9)

Básico de Teoria dos Números

- Inteiros relativamente primos
  - a e b são relativamente primos se mdc(a,b)=1
- Aritmética modular
  - a=qn+r, 0≤r<n; q=⌊a/n⌋
  - propriedades
    - a≡b mod n se n|(a-b)
    - se (a mod n)=(b mod n), então a≡b mod n
    - se a≡b mod n, então b≡a mod n
    - se a≡b mod n e b≡c mod n, então a≡c mod n

Básico de Teoria dos Números

- Aritmética modular (cont)
  - operações
    - [(a mod n)+(b mod n)] mod n = (a+b) mod n
    - [(a mod n)-(b mod n)]mod n = (a-b) mod n
    - [(a mod n)*(b mod n)]mod n = (a*b) mod n
- Função tociente de Euler
  - $\phi(n)$ é o número de inteiros menores que n e relativamente primos a n
  - para p primo, $\phi(p)=p-1$
    - se n=pq, p≠q, então $\phi(n)=(p-1)(q-1)$

Básico de Teoria dos Números

- Teorema de Euler
  - para todo a, n relativamente primos, $a^{\phi(n)} \equiv 1 \bmod n$
  - para p primo $a^p \equiv a \bmod n$
- Cálculo do inverso modular $a \equiv b^{-1} \bmod n$
  - algoritmo estendido de euclides
    - se mdc(n,a)=1, então o algoritmo estendido de Euclides proporciona o inverso modular de a mod n

Básico de Teoria dos Números

- Algoritmo estendido de Euclides
  - Exemplo: mdc(13,5)

i	r	q	s	t
-1	13		1	0
0	5		0	1
1	3	2	1	-2
2	2	1	-1	3
3	1	1	2	**-5**
4	0	2		

  - o número em negrito é o inverso modular de 5 mod 13, ou seja $5 \equiv -5^{-1} \bmod 13 \equiv 8 \bmod 13$

RSA

- Criado em 1977 por Rivest, Shamir e Adleman
- Bases matemáticas
  - dificuldade de fatorar o produto de dois números primos
  - inversibilidade da exponenciação em módulo, se usados determinados pares de números primos como expoentes
- Chave pública e chave privada
- Velocidade: 100 vezes mais lento que DES

RSA

- Encontrar 2 fatores primos do número 77
- Encontrar 2 fatores primos do número

  34.812.398.461.263.861.234.861.324.986.123.846.

  198.234.674.365.874.223.487.129.867.356.512.390.

  348.723.487.231.785.345.987.234.586.234.230.476.

  897.234.634.598.734.587.987.235.879.823.458.789.

  234.589.789.342.683
- Tempo de fatoração de um simples módulo de 200 dígitos RSA é maior que70,000,000 MIPS-years
- Mesmo nas máquinas mais rápidas, a vantagem é sempre do cifrador, pois chaves maiores podem ser usadas com maior eficiência do que a dos métodos de fatoração

RSA

- Chave pública
  - n produto de dois números primos, p and q
  - e primo relativo de (p-1) x (q-1)
- Chave privada
  - $n = p$ x q
  - $d = e^{-1} (\text{mod } (p\text{-}1) \text{ x } (q\text{-}1))$
- Cifração (usando a chave pública)
  - $c = m^e (\text{mod } n)$
  - e vale tipicamente 65536 (2^{16})
- Decifração (usando a chave privada)
  - $m = c^d (\text{mod } n)$
  - d vale de 2^{510} a 2^{512} para módulo de 512-bits
  - Operação mais demorada que a cifração

RSA

- Texto em claro é cifrado em blocos
  - cada bloco deve ter tamanho menor que $\log_2 n$, para algum inteiro n
  - na prática, o tamanho do bloco é 2^k, $2^k < n \leq 2^{k+1}$
- Forma da cifração e decifração:
  - $C = M^e \text{mod } n$
  - $M = C^d \text{mod } n = (M^e)^d \text{mod } n = M^{ed} \text{mod } n$

RSA

- Cálculo do par de chaves
  - Seja $\Phi(n)$ a função tociente de Euler (número de inteiros positivos menores que n e relativamente primos a n
  - Se $n=pq$, então $\Phi(n)=(p-1)(q-1)$
  - Pelo teorema de Euler: dados p e q primos, n e m inteiros, tq $n=pq$ e $0<m<n$, e um inteiro arbitrário k:

$$m^{k\Phi(n)+1}=m^{k(p-1)(q-1)+1}\equiv m \bmod n$$

  - Assim, $ed=k\Phi(n)+1$

RSA

- Esquema RSA:
  - p, q dois números primos (privado, escolhido)
  - n=pq (público, calculado)
  - e, com $mdc(\Phi(n),e)=1$; $1<e<\Phi(n)$ (público, escolhido)
  - $d\equiv e^{-1} \bmod \Phi(n)$ (privado, calculado)
- Assim,
  - KU={e,n} e KR={d,n}

RSA

- Segurança do RSA
  - Possibilidades de ataques:
    - força bruta
    - ataques matemáticos
      - fatorar n em seus dois fatores primos
      - determinar Φ(n) diretamente
      - determinar d diretamente
    - ataques de temporização

RSA

- Progresso na fatoração

Nº dígitos decimais	Data	MIPS-ano	Método
100	1991	7	MPQS
110	1992	75	MPQS
120	1993	830	MPQS
129	1994	5000	MPQS
130	1996	500	NFS
140	1999	2000	NFS
155	1999	8000	NFS

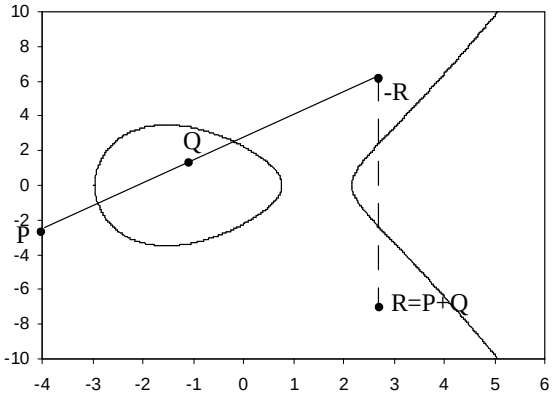
- RSA Challenge: http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html

RSA

- Ataques de temporização
  - Baseado em diferenças de tempo no cálculo de uma operação (por exemplo: exponenciação modular) com operandos diferentes
  - Formas de se evitar:
    - tempo de exponenciação constante
    - atrasos aleatórios
    - cegueira: multiplicação por um aleatório r

Criptografia de Curva Elíptica

- CE no corpo dos reais: Forma de Weierstrass:
  $y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6$



Criptografia de Curva Elíptica

– A curva elíptica *E* é composta por todos os pontos *(x,y)*, *x* e *y* ∈ *K*, que satisfazem a definição da curva, juntamente com o ponto no infinito *O*.

Criptografia de Curva Elíptica

– Regras de adição:

P+O=O+P=P

(x,y)+(x,-y)=O

P=(x_1,y_1) e *Q=(x_2,y_2)*. Então, *P+Q=(x_3,y_3)*, onde:

- $x_3=\lambda^2-x_1-x_2$
- $y_3=\lambda(x_3-x_1)+y_1$
- $\lambda=(y_2-y_1)/(x_2-x_1)$, *se* $P\neq Q$
- $\lambda=(3x_1^2+a)/2y_1$, *se* $P=Q$

– Regra de multiplicação: refere-se ao produto entre um escalar *k* e um ponto sobre a curva *P*

- *Q=kP*

Criptografia de Curva Elíptica

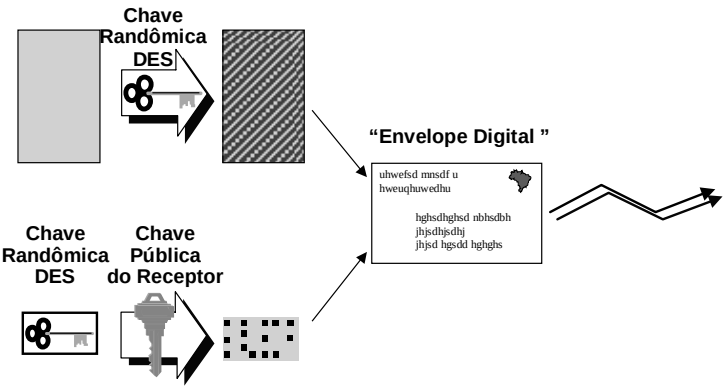
- Cifração e decifração
  - embutir dados na curva
    - $P_A=k_AG$
    - $P_B=k_BG$
    - Alex envia $C_m=\{kG,P_m+kP_B\}$
    - Bia calcula

      $(P_m+kP_B)-k_B(kG)=P_m+k(k_BG)-k_B(kG)=P_m$

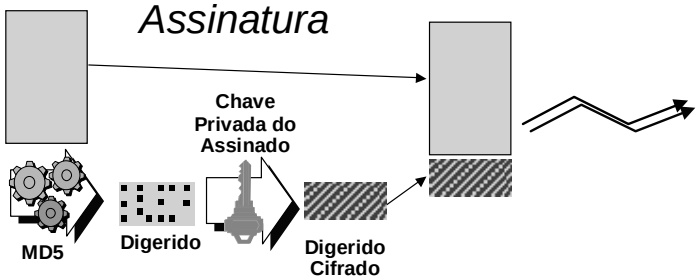
Criptografia de Curva Elíptica

Alex		Comum	Bia	
		Tamanho do corpo		
		Ponto Base G		
Privado Alex	Público Alex	Curva E	Público Bia	Privado Bia
Chave privada k_A	Chave pública $P_A=k_AG$	Base matemática	Chave pública $P_B=k_BG$	Chave privada k_B

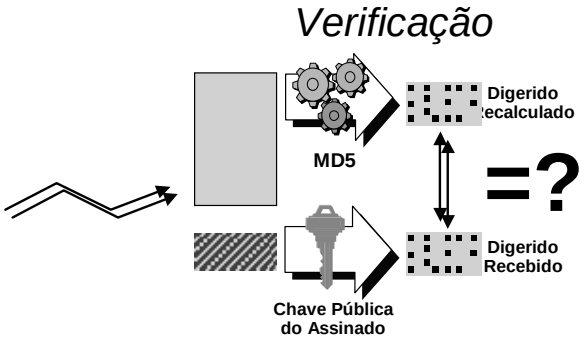
Combinações: Envelope Digital



Combinações: Assinatura Digital



Combinações: Assinatura Digital



Combinações: Assinatura Digital

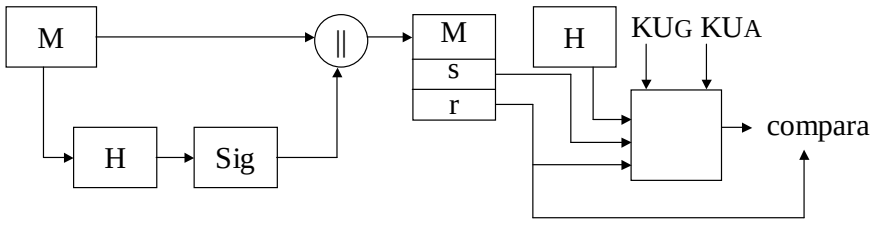
- Requisitos:
  - depende da mensagem
  - utiliza informação única da fonte
  - fácil de produzir
  - fácil de reconhecer
  - intratável de forjar
  - armazenamento prático

Combinações: Assinatura Digital

- Abordagens
  - assinatura direta
  - assinatura arbitrada
    - usando criptografia simétrica
      - (1) X→A: M || $E_{K_{XA}}$[ID$_X$ || H(M)]
      - (2) A→Y: EK_{XB}[ID$_X$ || M || $E_{K_{XA}}$[ID$_X$ || H(M)] || T]
    - chave secreta compartilhada previamente entre X e Y
      - (1) X→A: ID$_X$ || $E_{K_{XY}}$[M] || $E_{K_{XA}}$[ID$_X$ || H($E_{K_{XY}}$[M])]
      - (2) A→Y: $E_{K_{AY}}$[ID$_X$ || $E_{K_{XY}}$[M] || $E_{K_{XA}}$[ID$_X$ || H($E_{K_{XY}}$[M]) || T]
    - usando criptografia assimétrica
      - (1) X→A: ID$_X$ || E_{KR_X}[ID$_X$ || E_{KU_Y}(E_{KR_X}[M])]
      - (2) A→Y: E_{KR_A}[ID$_X$ || E_{KU_Y}(E_{KR_X}[M]) || T]

Combinações: Assinatura Digital

- Digital Signature Standard (DSS)



Gerenciamento de Chaves

- Chaves privadas
  - Muito grandes, não permitem memorização
  - Necessidade de proteção para mantê-las privadas
- Chaves Públicas
  - Devem ser amplamente divulgadas
  - Conveniente ligá-las a um nome
- Dilema da incerteza sobre identidade do dono de uma chave
  - Qualquer um pode gerar um par de chaves pública/privada
  - Qualquer um pode dar um nome a uma chave pública
  - Qualquer um pode divulgar uma chave pública em um diretório público
  - Dilema: Como saber *com certeza* que o nome em uma chave pública representa realmente o receptor com quem se quer comunicar?
  - Solução: Certificados Digitais

Gerenciamento de Chaves

- Considerações:
  - distribuição de chaves públicas
  - uso de criptografia de chave pública para distribuição de chaves secretas (key exchange)

Gerenciamento de Chaves

- Distribuição de chaves públicas
  - Esquemas
    - anúncio público (ex: PGP)
    - diretório disponível publicamente
    - autoridade de chave pública
    - certificados de chave pública

Gerenciamento de Chaves

- Distribuição de chaves secretas
  - Esquemas
    - Distribuição simples de chave secreta
      - Alex gera par de chaves
      - Bia cifra segredo com chave pública de Alex
      - Alex decifra
    - Distribuição com confidencialidade e autenticação
    - Esquema Híbrido
      - Uso de um KDC (key distribution center)

Diffie-Hellman

- Problema comum a DES, RC2, RC4: acordo prévio entre emissor e receptor quanto à chave de criptografia
- Algoritmo Diffie-Hellman
  - permite aos dois lados derivar uma chave sem necessidade de trocar nenhuma informação secreta
  - tem base na dificuldade de calcular logaritmos discretos
  - troca de chave simétrica
  - vulnerabilidade *man-in-the-middle*
- Troca de chave através de um canal não-seguro
  - Confidencialidade garantida
  - Autentificação não garantida

Diffie-Hellman

- Baseado na dificuldade de se calcular logaritmos discretos
- A Função Logaritmo Discreto é inversa da função exponenciação modular, definida como $y=g^x \bmod p$, onde p é primo e g é um gerador de Z/pZ

Diffie-Hellman

	A		B
1.	p,g	→	p,g

p = número primo, g = base do logaritmo

2. $y_1 = g^{x1} \bmod p$ $\quad y_2 = g^{x2} \bmod p$

x_1 = valor privado da 1a. parte
x_2 = valor privado da 2a. parte

Diffie-Hellman

	A		B
1.	p,g	→	p,g
2.	$y_1 = g^{x1} \bmod p$		$y_2 = g^{x2} \bmod p$
3.	y_2	← →	y_1

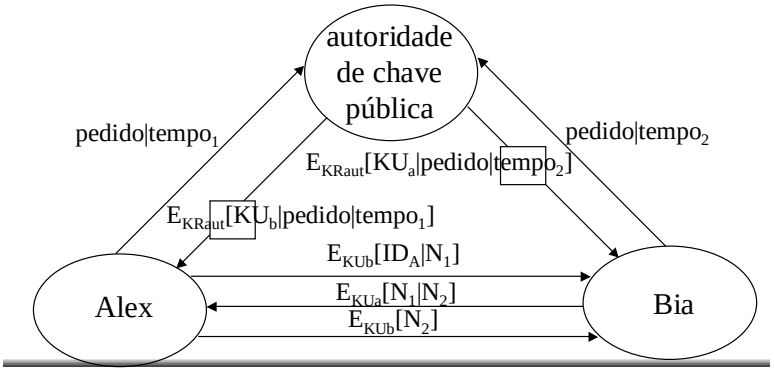
y_1 = valor público da 1a. parte
y_2 = valor público da 2a. parte

4. $K = y_2^{x1} \bmod p$ $\quad K = y_1^{x2} \bmod p$

K = chave secreta

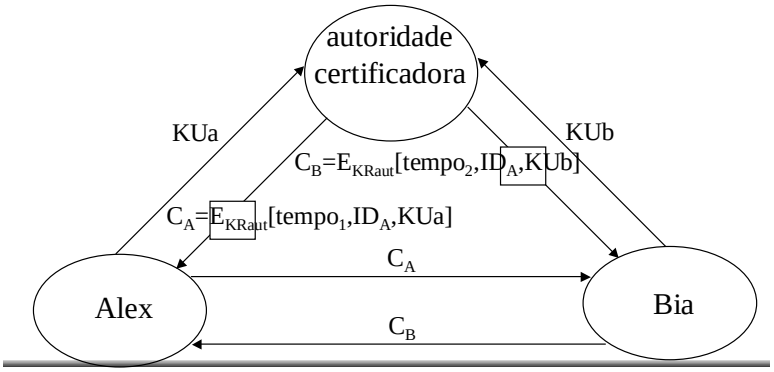
Gerenciamento de Chaves

- Autoridade de chave pública



Gerenciamento de Chaves

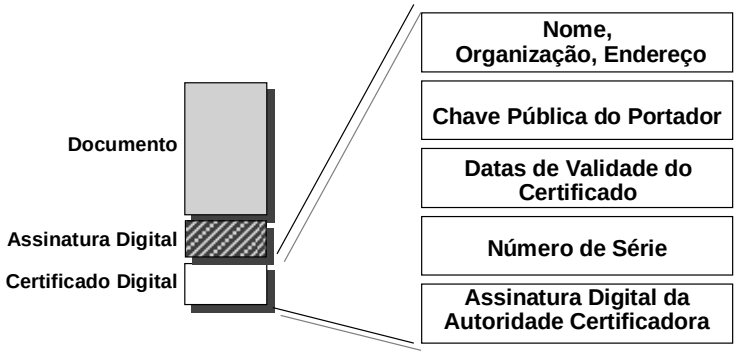
- Certificados de chave pública



Certificados Digitais

- Características
  - Como uma carteira de identidade eletrônica, um Certificado Digital assegura que alguma autoridade fez checagens razoáveis da identidade do portador
  - Um Certificado Digital notariza a ligação entre uma chave pública e um Emissor, assim como o R.G. prova a identidade de alguém

Certificados Digitais



Autoridades Certificadoras

- São o equivalente eletrônico dos cartórios (notários): "terceira parte confiável"
- Emitem, autenticam e gerenciam certificados digitais
- Serviço contínuo, confiável, operando 24 horas por dia, 7 dias por semana, e acessíveis globalmente pelos clientes
- Recursos tecnológicos
  - protocolos e padrões de segurança
  - sistema de mensagens seguro
  - infraestrutura adequada: instalações seguras, banco de dados, *home-page*
  - serviço de suporte ao cliente
- Conjunto de *práticas*:
  - normas e procedimentos de utilização dos serviços pelos clientes
  - resolução de disputas

Infra-Estrutura de Chave Pública