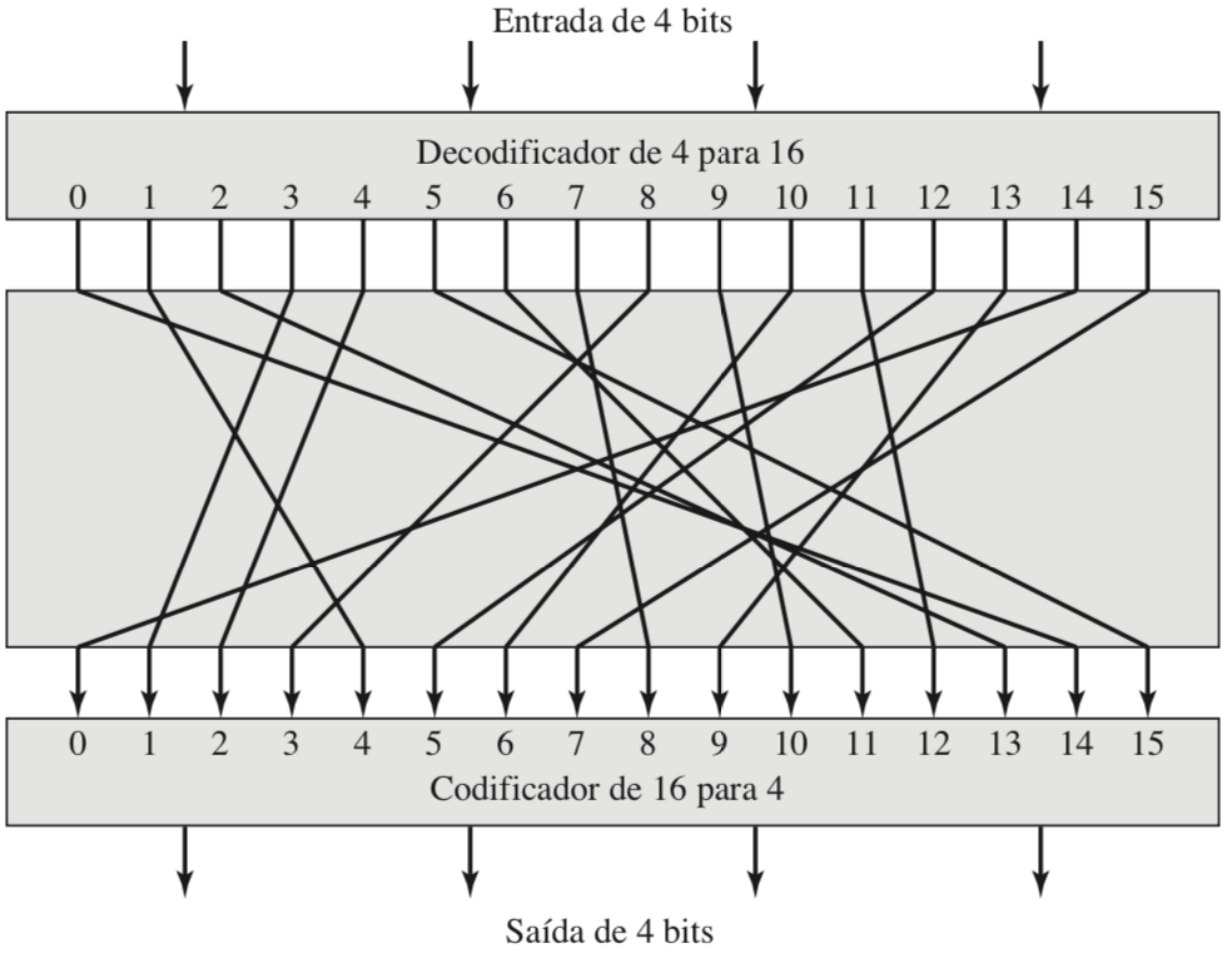
Segurança da Informação– GBC083

Prof. Rodrigo Sanches Miani – FACOM/UFU

Aula passada

Segurança da Informação– GBC083

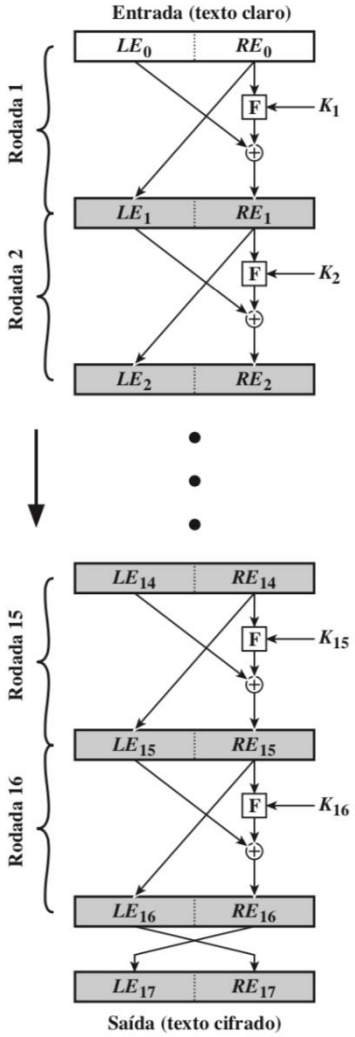
Cifra de substituição geral – n=4



Cifra de substituição geral – Problemas...

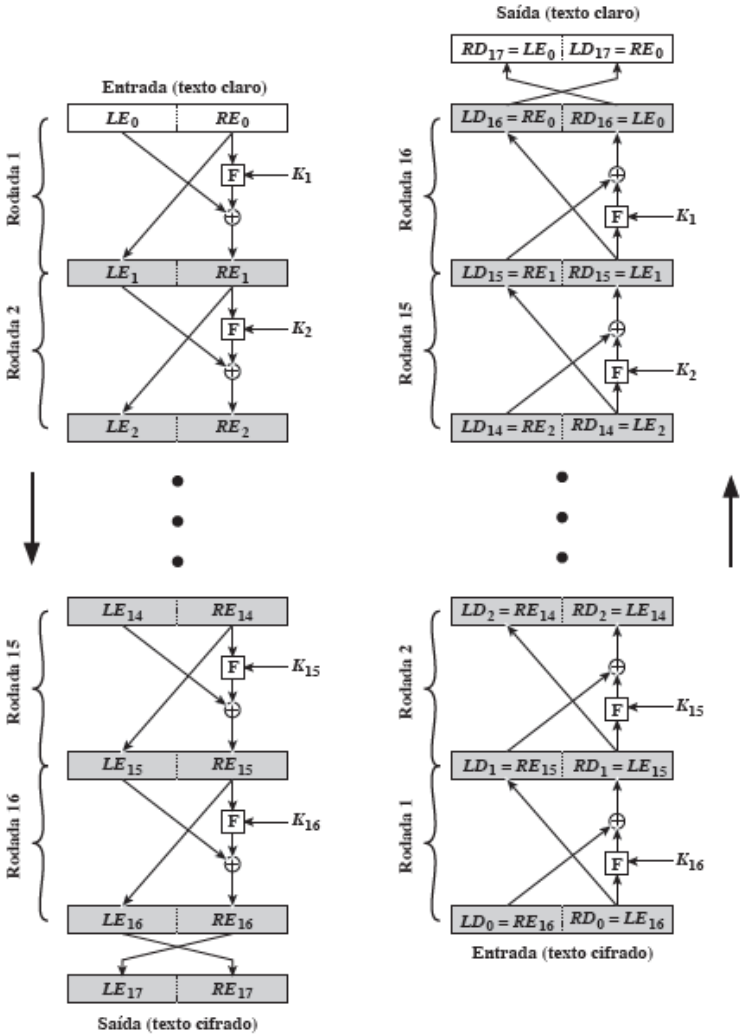
- A ideia, então, seria aumentar o tamanho de n!
- Qual seria o tamanho da nossa chave para o caso com n=4?
  - Basta multiplicar 4 (bits) x 2^4 (linhas) = 64 bits (literalmente o tamanho da cadeia de bits da tabela anterior);
- Como ficaria a conta para um tamanho de bloco n "razoável"?
  - Com n = 64 teríamos: 64 x 2^64 = 10^12 bits!!! Isso é prático?

Cifra de Feistel – Cifragem (*encryption*)



- Entradas do algoritmo:
  1. Um bloco de texto claro de tamanho 2w bits;
  2. Uma chave K.
- Funcionamento básico:
  - O bloco do texto claro é dividido em duas metades L_0 e R_0 (cada uma com w bits);
  - As duas metades dos dados passam por n rodadas de processamento e depois se combinam para produzir o bloco do texto cifrado;
  - Cada rodada i possui como entradas L_{i-1} e R_{i-1}, derivadas da rodada anterior, assim como uma subchave K_i derivada de K;

DES – Cifrar/decifrar



Tópicos da aula

1. Ideia geral
2. Geração das subchaves
3. Descrição de uma rodada e da função F
4. Discussão
5. Princípios de projeto de cifra de bloco

DES – Ideia geral

Segurança da Informação– GBC083

DES – *Data Encryption Standard*

- Cifra adotada pelo governo americano em 1977;
- Desenvolvida pela IBM;
- Gerou muita controvérsia por ser baseada em uma cifra patenteada pela IBM;
  - Havia a suspeita que o governo americano tinha uma *backdoor*.
- Até 1999 era o algoritmo simétrico padrão adotado pelo governo norte-americano e em diversos protocolos criptográficos (SSL, por exemplo).

DES

- Veremos que, em sua forma original, o DES não é mais seguro;
- Chave de 64 bits é muito pequena;
- Ainda em 1977, Diffie e Hellman **projetaram** uma máquina que poderia decifrar o DES.

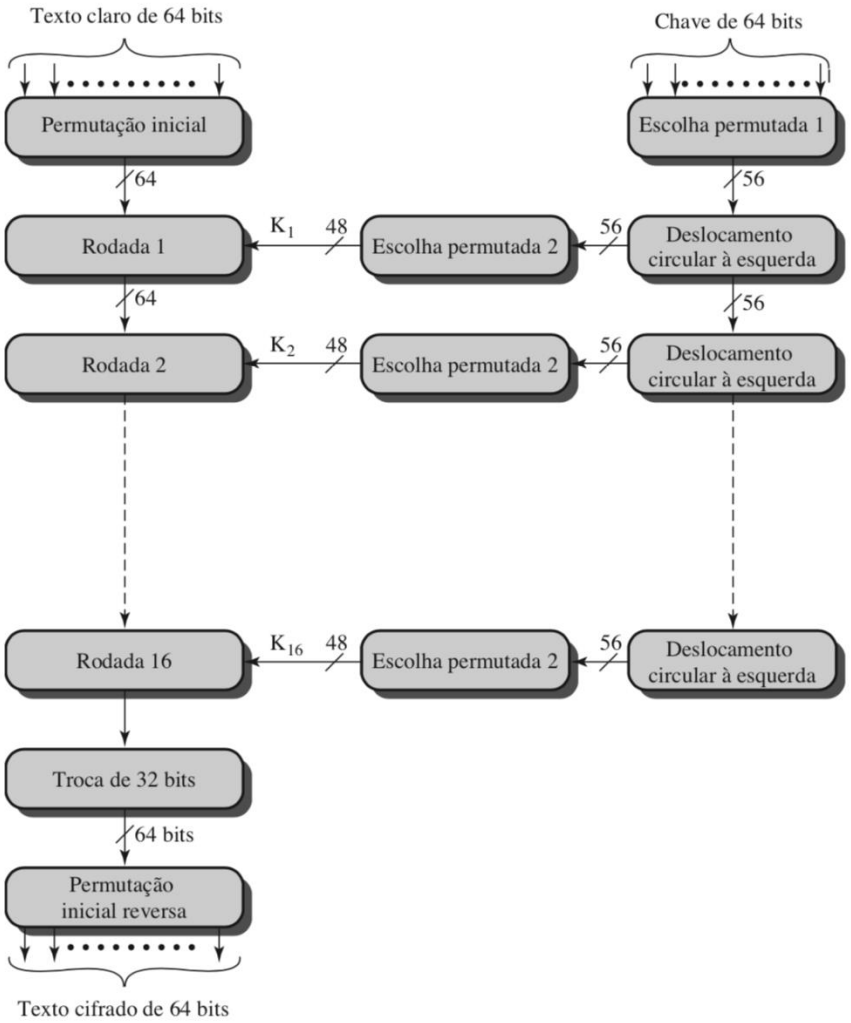
DES – ideia básica

- Entrada
  1. Texto claro de 64 bits
  2. Chave de 64 bits

- Usando uma rede de Feistel com uma série de substituições e permutações, o algoritmo transforma os 64 bits de entrada em uma saída de 64 bits (texto cifrado).

- As mesmas etapas, com a mesma chave, são usadas para obter o texto claro à partir do texto cifrado.

DES – representação geral



DES – diferenças com a cifra de Feistel

- Com exceção das permutações inicial (IP) e final (IP^{-1}), o DES tem a estrutura exata de uma cifra de Feistel;
  - Já conhecemos a estrutura de uma cifra de Feistel!

- Dois itens ainda não foram esclarecidos:
  1. Geração das subchaves;
  2. Conteúdo da função F.

DES – Geração das subchaves

Segurança da Informação– GBC083

DES – Geração das subchaves

- As subchaves são necessárias por dois motivos:
  1. Projeto do algoritmo (cifra de produto) envolve uma série de rodadas;
  2. Usar chaves repetidas (ou até a mesma chave!) em cada uma das rodadas do algoritmo enfraquece a ideia de tornar obscura a relação entre texto claro e texto cifrado.

DES – Geração das subchaves

- Algoritmo
  1. Entrada de 64 bits -> K
  2. Escolha permutada de 56 bits -> K_p
  3. Deslocamento circular à esquerda em K_p -> K_{pe}
  4. Escolha permutada de 48 bits em K_{pe} -> K_1

DES – Geração das subchaves (1)

- **K** = 0001 0011 0011 0100 0101 0111 0111 1001 1001 1011 1011 1100 1101 1111 1111 0001

DES – Geração das subchaves (2)

- **K** = 00010011 00110100 01010111 01111001 10011011 10111100 11011111 11110001
  - Escolha permutada usando a seguinte tabela:

PC-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

- $\mathbf{K_p}$ = 11111000 01100111 00101010 01011111 01010101 10111001 10011111 00011111

DES – Geração das subchaves (3)

- $\mathbf{K_p}$ = 1111000 0110011 0010101 0101111 0101010 1011001 1001111 0001111
- $\mathbf{K_p}$ é dividida em duas metades – C_0D_0;
- Cada subchave K_n será gerada à partir de deslocamentos usando a tabela ao lado.
  - Mova o bit para esquerda, com exceção do primeiro que irá para o fim do bloco.

Iteration Number	Number of Left Shifts
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

DES – Geração das subchaves (3)

$C_0 = 11110000110011001010101011111$

$D_0 = 01010101011001100111100011111$

$C_1 = 11100001100110010101011111$

$D_1 = 10101010110011001111100011110$

Iteration Number	Number of Left Shifts
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

DES – Geração das subchaves (4)

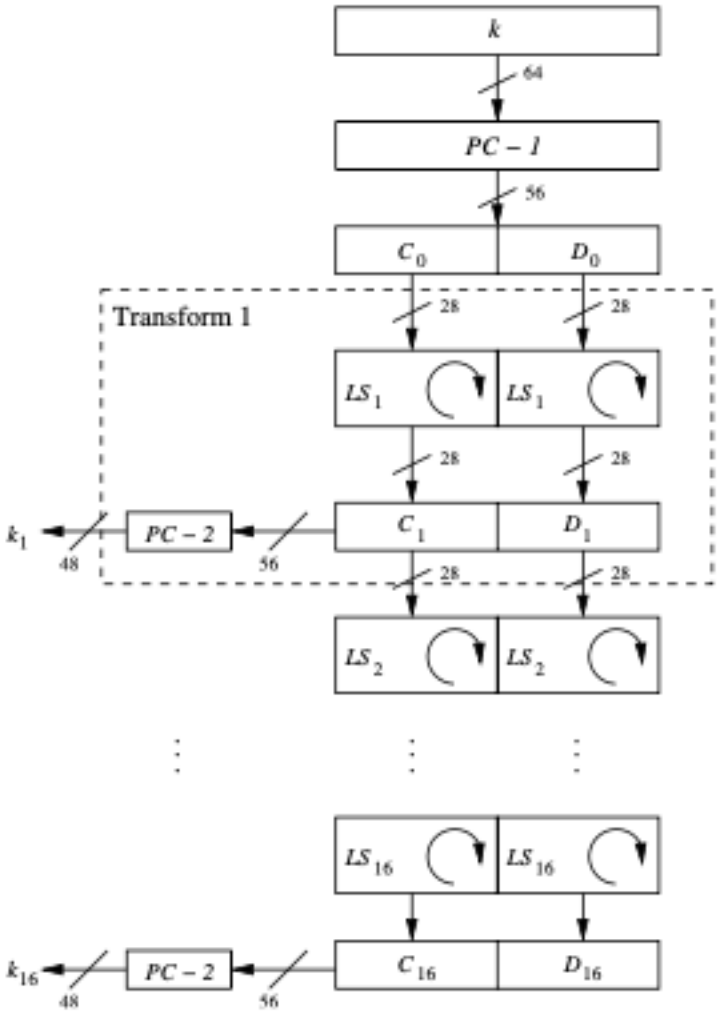
- As subchaves estão quase prontas...
- Basta agora selecionar uma permutação delas usando a tabela ao lado;
- Se C_1D_1 = 1110000 1100110 0101010 1011111 1010101 0110011 0011110 0011110 então:
- K_1 = 000110 110000 001011 101111 111111 000111 000001 110010

- Qual é o tamanho das subchaves?

PC-2

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

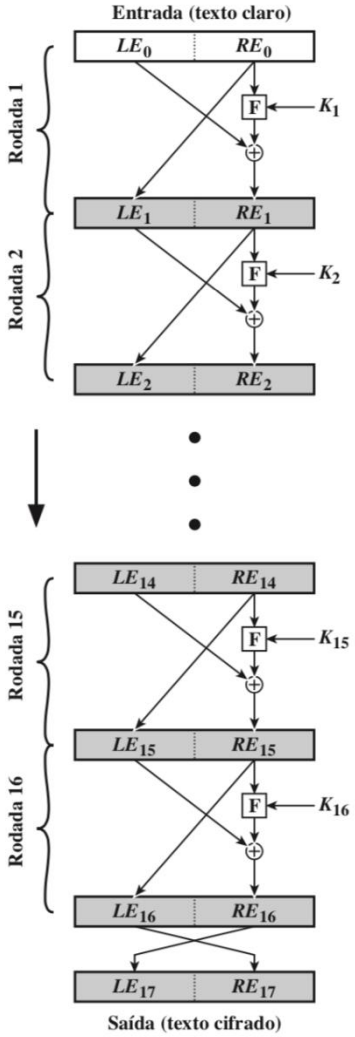
DES – Geração das subchaves – Visão geral



DES – Descrição de uma rodada e da função F

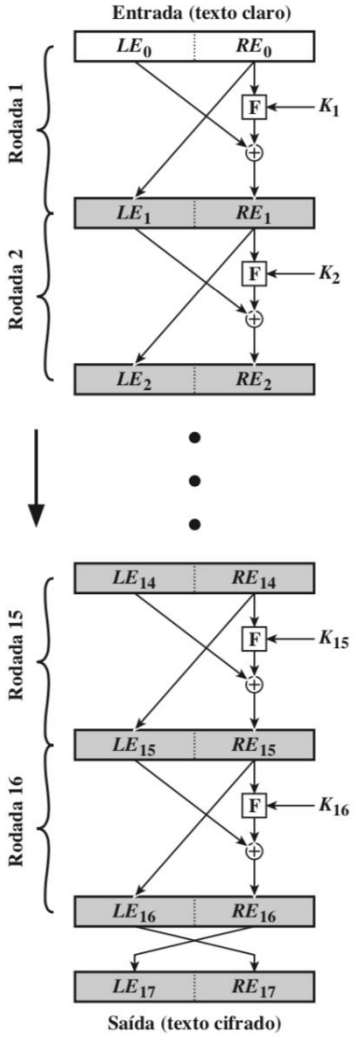
Segurança da Informação– GBC083

DES – Rodadas e função F



- Entradas do algoritmo:
  1. Um bloco de texto claro de tamanho 64 bits;
  2. Uma chave K de 64 bits.
- Funcionamento básico:
  - O bloco do texto claro é dividido em duas metades L_0 e R_0 (cada uma com 32 bits);
  - As duas metades dos dados passam por 16 rodadas de processamento e depois se combinam para produzir o bloco do texto cifrado;
  - Cada rodada i possui como entradas L_{i-1} e R_{i-1}, derivadas da rodada anterior, assim como uma subchave K_i derivada de K;
  - Cada subchave possui 48 bits.

DES – Rodadas e função F



- Permutação inicial (IP)

IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

O 58th bit da mensagem M original será agora o primeiro bit da mensagem M permutada e assim por diante...

DES – IP e IP^{-1}

- O intuito dessas tabelas nunca foi muito claro...
- Ela acontece antes de usar a chave e pode ser desfeita por qualquer um!
- Vale muito a pena olhar o porquê aqui:
  - https://crypto.stackexchange.com/questions/3/what-are-the-benefits-of-the-two-permutation-tables-in-des

The Initial Permutation: IP

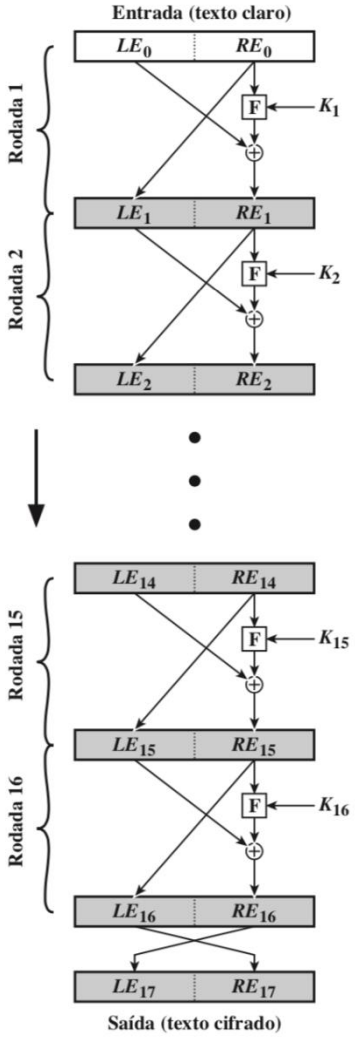
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

DES – Rodadas e função F

A função rodada F é o coração de um algoritmo de criptografia simétrica;

- O DES, por exemplo, faz:
  - i) uma **permutação (difusão)** inicial: divide os bits em oito grupos de seis bits;
  - ii) uma **substituição (confusão)** em cada um dos grupos para produzir grupos de quatro bits;
  - iii) usa uma tabela de **permutação (difusão)** em cada um dos oito grupos de quatro bits.

DES – Rodadas e função F

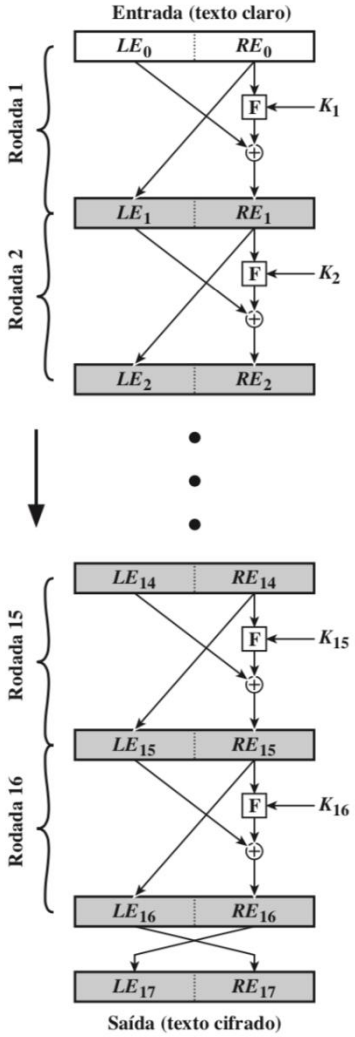


- Antes de RE_0 entrar na função F é preciso realizar uma expansão de bits;
- RE_0 tem 32 bits e precisará aumentar para 48 bits (tamanho da subchave);

E BIT-SELECTION TABLE

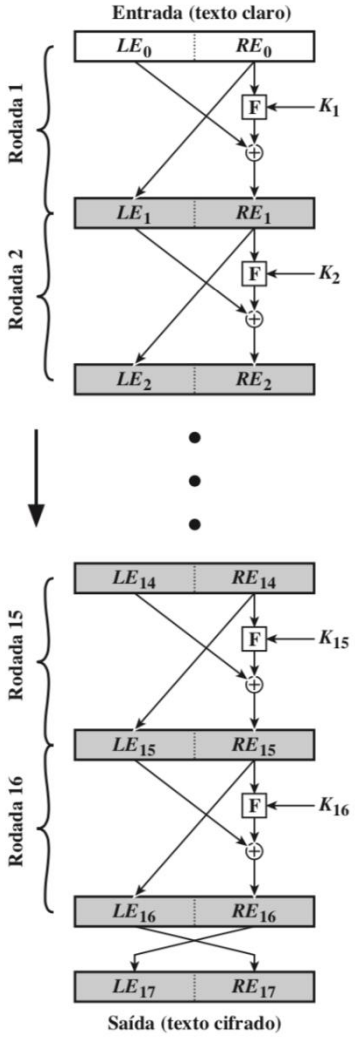
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

DES – Rodadas e função F



- O primeiro passo feito dentro de F é fazer um XOR do bloco RE_0 estendido com a subchave K_1;
- O próximo passo, diminuir de 48 bits para 32 bits, envolve a aplicação de oito S-boxes.

DES – Rodadas e função F



- Para diminuir de 48 bits para 32 bits, primeiramente os 48 bits são organizados em oito grupos de seis bits;
- A ideia é transformar esses seis bits em quatro bits e aí eu voltaria a ter os 32 bits que preciso para o próximo passo (XOR com a LE_0);
- Isso será feito usando o conceito de S-box.

DES – Rodadas e função F

S1

Row No.	Column Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

- A tabela acima ilustra a primeira S-box, chamada de S1 (no total são oito!);
- Ela será aplicada aos primeiros 6 bits;
- Exemplo:
  - Objetivo: 6 bits "viram" 4 bits;
  - S1(011011) = 0101 (primeiro e último bit fornecem a linha, os quatro bits do meio fornecem a coluna;

DES – Rodadas e função F

S1

Row No.	Column Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

- A tabela acima ilustra a primeira S-box, chamada de S1 (no total são oito!);
- Ela será aplicada aos primeiros 6 bits;
- Exemplo:
  - Objetivo: 6 bits "viram" 4 bits;
  - S1(011011) = 0101 (primeiro e último bit fornecem a linha, os quatro bits do meio fornecem a coluna;
  - Linha – "01" = 1 e Coluna "1101" = 13, portanto, 5 = "0101".

DES – Rodadas e função F

- Após a execução das oito S-boxes o algoritmo terá um conjunto de 32 bits que é uma mistura da subchave com diversas substituições;
- O último passo da função F é justamente uma permutação em tais 32 bits;
- A saída da função F será combinada com LE_0 usando um XOR. Tal saída será o RE_1.

P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

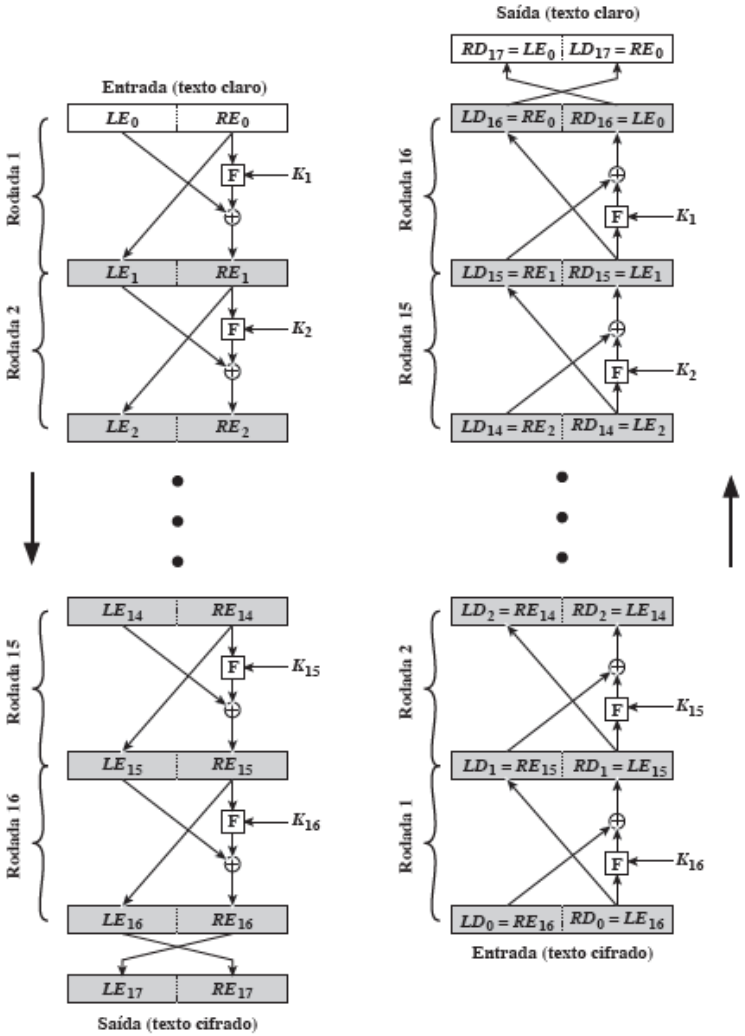
DES – Rodadas e função F

- As rodadas são executadas levando em consideração as subchaves calculadas e que $LE_{i+1} = RE_i$ ou seja, $LE_2 = RE_1$ e assim sucessivamente...
- A última rodada irá produzir o $LE_{16}RE_{16}$;
- Duas últimas permutações são feitas aqui:
  - Trocar RE com LE;
  - Usar uma última tabela de permutação – IP^{-1}

DES – Discussão

Segurança da Informação– GBC083

DES – Cifrar/decifrar



DES – Discussão

- Dado um texto cifrado com DES, quais seriam as opções para um criptoanalista?
  - Como o tamanho do bloco é de 64 bits, acertar o texto claro envolve percorrer as 2^64 possibilidades. O quão difícil é isso?
  - Dado que o algoritmo é público, o quão difícil seria fazer o processo inverso?

DES – Segurança

Dois pontos são importantes:

1. Tamanho da chave
2. Natureza do algoritmo

DES – Segurança (tamanho da chave)

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

DES – Segurança (natureza do algoritmo)

- Outra preocupação é a possibilidade de que a criptoanálise seja possível explorando-se as características do algoritmo DES;
- O foco disso tem sido as oito tabelas de substituição, ou S-boxes;
- Como os **critérios** de projeto para essas caixas, e, na realidade, para o algoritmo inteiro, não se tornaram públicos, existe uma suspeita de que elas foram construídas de modo que a criptoanálise seja possível para um oponente que conheça as fraquezas nelas...

Triplo DES

- Para superar os problemas de segurança do DES, em 1979 foi criado o Triplo DES.
- O principal objetivo foi aumentar o tamanho da chave de segurança.
- Manteve a compatibilidade com sistema que usavam o DES.
- Utiliza três chaves de 56 bits:
  - Cifra com a chave 1, decifra com a chave 2 e cifra com a chave 1 (56*3 = 168 bits).

Triplo DES - Adoção

- O 3DES ou (TDEA) ainda é utilizado em aplicações pelo mundo;
- OpenPGP é uma delas – protocolo para cifragem de email;
- Contudo, diversas vulnerabilidades foram encontradas no 3DES, especialmente relacionadas ao tamanho máximo do bloco que é cifrado – 64 bits;
- O NIST já divulgou uma nota em que afirma que já está preparando a descontinuidade do 3DES:
  - https://csrc.nist.gov/CSRC/media/Publications/Shared/documents/itl-bulletin/itlbul2017-11.pdf

DES – Princípio de projeto de cifra de bloco

Segurança da Informação– GBC083

Princípios de projeto de cifra de bloco

Embora tenha havido muito progresso no projeto de cifras de bloco criptograficamente fortes, os princípios básicos não mudaram tanto desde o trabalho de Feistel e da equipe de projeto do DES, no inicio da década de 1970.

Três aspectos críticos para o projeto de cifra de bloco são:

1. Número de rodadas;
2. Projeto da função F;
3. Algoritmo de geração de subchave.

Número de rodadas

- Quanto maior o número de rodadas, mais difícil é realizar a criptoanálise, mesmo para uma função F relativamente fraca;

- Em geral, o critério deverá ser de que o número de rodadas seja escolhido de modo que os esforços criptoanalíticos conhecidos exijam maior ação do que um ataque de busca de chave por força bruta.

Número de rodadas

- Análises no DES mostram que determinados tipos de ataques de criptoanálise são menos eficientes do que a força-bruta...
- Se o DES tivesse 15 ou menos rodadas a criptoanálise exigiria menos esforço do que a força-bruta;
- Esse é um critério interessante para avaliar e calcular o número de rodadas de um algoritmo simétrico de bloco.

Projeto da função F

- O núcleo da cifra de bloco de Feistel é a função F, que oferece a propriedade de **confusão** em uma cifra de Feistel;
- Ou seja, é preciso que seja difícil "desembaralhar" a substituição realizada por F;
  - Efeito avalanche;
    - Uma mudança em um bit da entrada deverá produzir alterações em muitos bits de saída.
- Em linhas gerais, quanto menos linear for F, mais difícil será qualquer tipo de criptoanálise...

Projeto da função F

- Uma propriedade desejável em algoritmos criptográficos é chamada de **efeito avalanche;**

- A ideia dessa propriedade é a de que uma pequena modificação na entrada (como a troca de um único bit, por exemplo), deve fazer com que a saída do algoritmo seja significantemente diferente.

Princípios de projeto de cifra de bloco – algoritmo de escalonamento de chave

- Selecionar subchaves de forma a maximizar a dificuldade de:
  - deduzir subchaves individuais;
  - recuperar a chave principal;

- O escalonamento de chave deve permitir a avalanche de bits.

Próximas aulas

- Outro algoritmo de bloco – AES (substituto do DES);
- Modos de operação de cifra de bloco.

Roteiro de estudos

1. Leitura das seções 3.2, 3.3, 3.4 e 3.5 do livro "Criptografia e segurança de redes. Princípios e práticas". William Stallings;
2. Estudo das vídeo-aulas referentes ao tópico 6 (parte 1 e parte 2);
3. Resolução do TP3.
4. "Brincar" com o DES usando alguma implementação disponível na Internet, por exemplo:
   - https://anycript.com/crypto/des
   - https://emvlab.org/descalc/