Tópicos em Redes de Computadores

Criptografia de Chave Secreta

Prof. Elias P. Duarte Jr.
Universidade Federal do Paraná (UFPR)
Departamento de Informática
www.inf.ufpr.br/elias/topredes

Sumário

- O que é criptografia de chave secreta?
- Circuitos para criptografia
- O Algoritmo DES
- O Algoritmo AES
- O Algoritmo IDEA
- Modos de uso dos algoritmos
- Conclusões

Criptografando

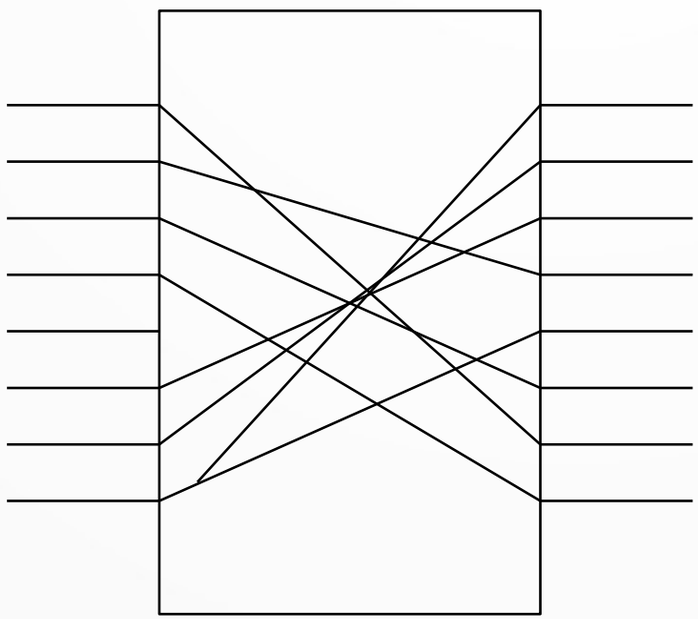
- Ana criptografa para Beto:
  - *O mapa está debaixo do tapete.*
- Fulana tenta espionar a mensagem:
  - **K%$loidjfnv@poi()jalsdkfirn*
- Beto sabe como descriptografar:
  - *O mapa está debaixo do tapete.*

Algoritmos Públicos & Chaves Secretas

- Texto Entrada <-> Algoritmo(k) <-> Texto Criptografado
- Apenas emissor e receptor podem conhecer a chave k
- Baseados nos princípios básicos de criptografia: substituição & transposição
- Os algoritmos são conhecidos, mas deve ser difícil obter o texto original sem a chave

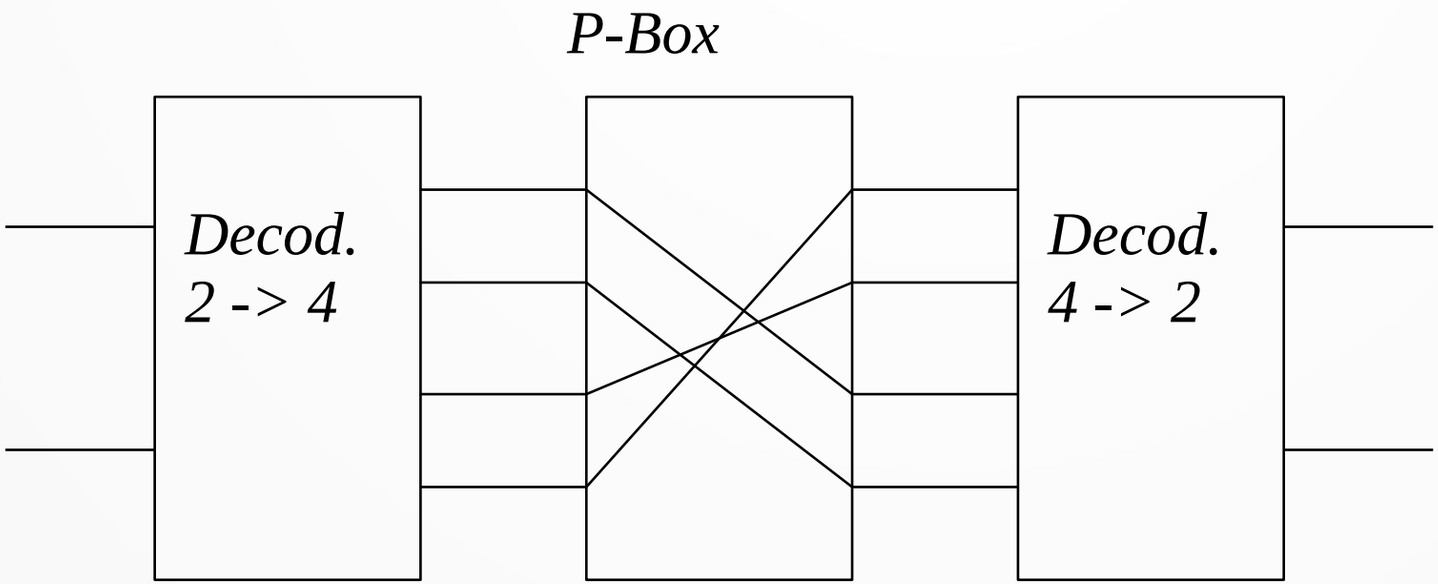
Circuitos para Criptografia: Transposição

- P-Box (***P****ermutation* ***Box***)



Circuitos para Criptografia: Substituição

- S-Box (***S**ubstitution **Box***)



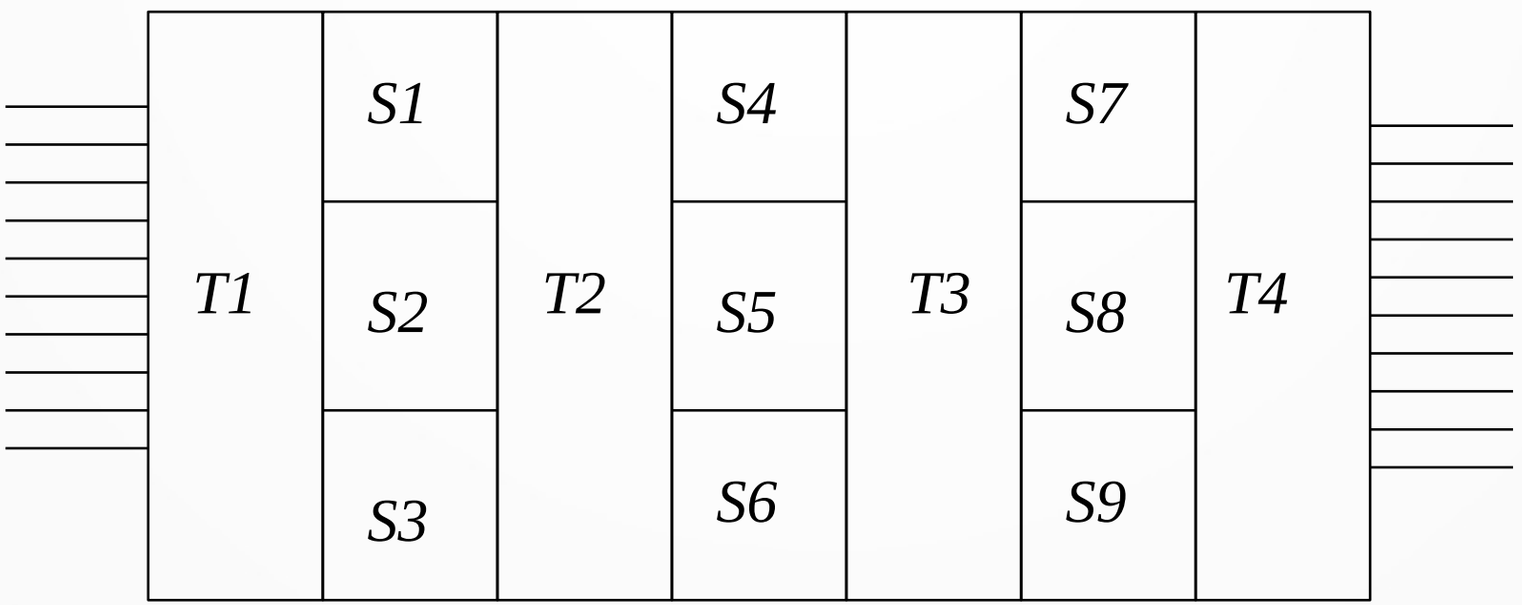
Exercício S-Box

- No circuito anterior, qual para qual valor o número $3_{10} = 11_2$ é mapeado?

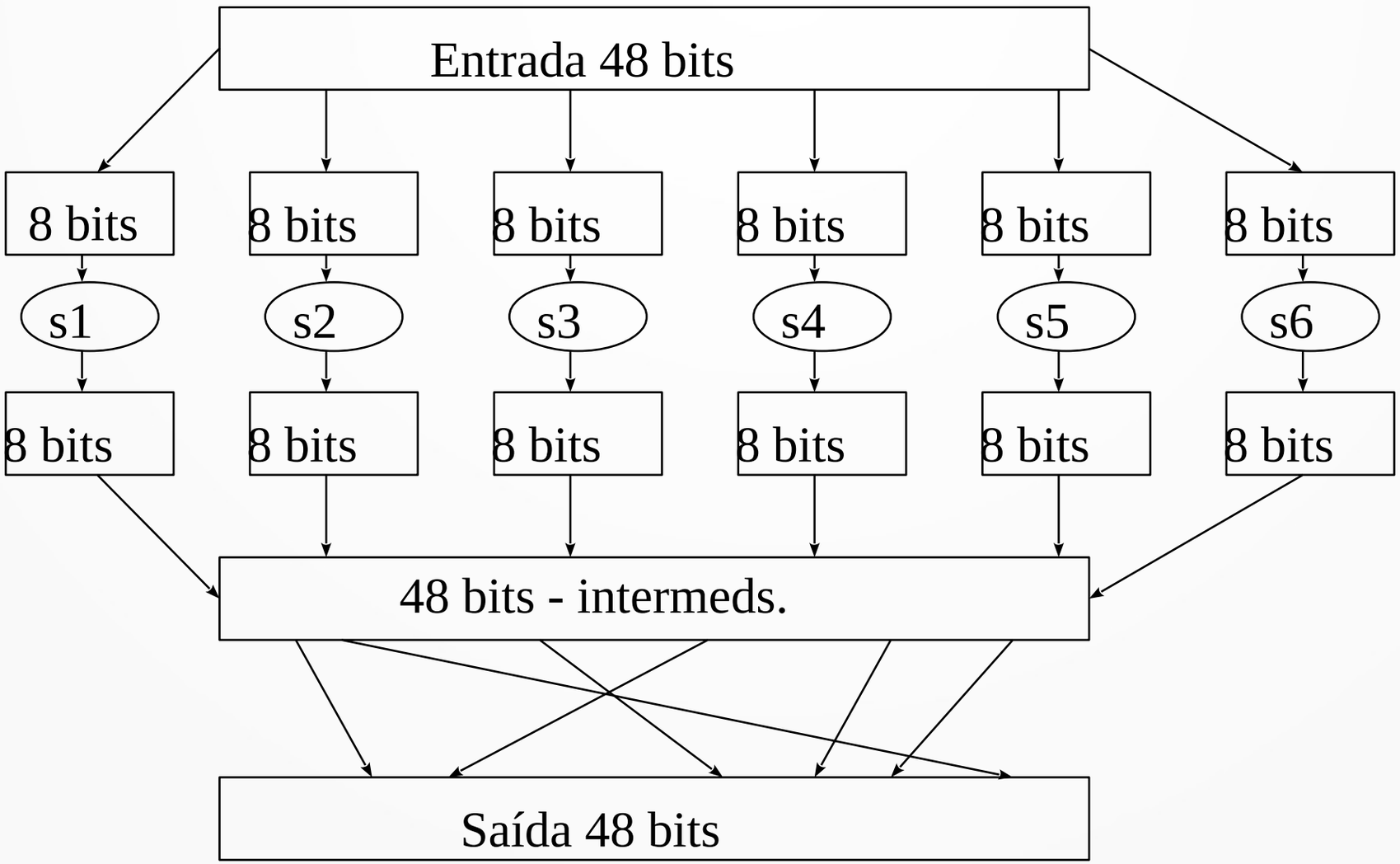
Exercício S-Box

- No circuito anterior, qual para qual valor o número $3_{10} = 11_2$ é mapeado?
- Resposta: inicialmente para 0001 → substituição → 1000 e então → para o número $00_2 = 0_{10}$
- Atenção: são 2^i alternativas
- Ex.: 2 bits na entrada → 4 alternativas; 10 bits na entrada → 1024 alternativas, crescimento exponencial!

Combinações em Cascata



Criptgrafando Blocos



O Algoritmo DES

- DES: *Data Encryption Standard*
- Padrão do governo dos EUA para uso comercial e não-militar
- Entrada: bloco de 64 bits
- Saída: bloco de 64 bits criptografados
- Chave de 56 bits + 1 bit paridade por byte
- Eficiente para implementação em hardware
- Nem tanto para implementação em software
- Uma CPU de 50 MIPS criptografa 300KBps

DES: Chave de 56 Bits

- Por que apenas 56 bits?
- O DES se origina do algoritmo Lucifer da IBM, com chave que 128 bits
- Já foi alegado que, além de diminuir a chave, o algoritmo foi enfraquecido
- Qual a razão dos bits de paridade?

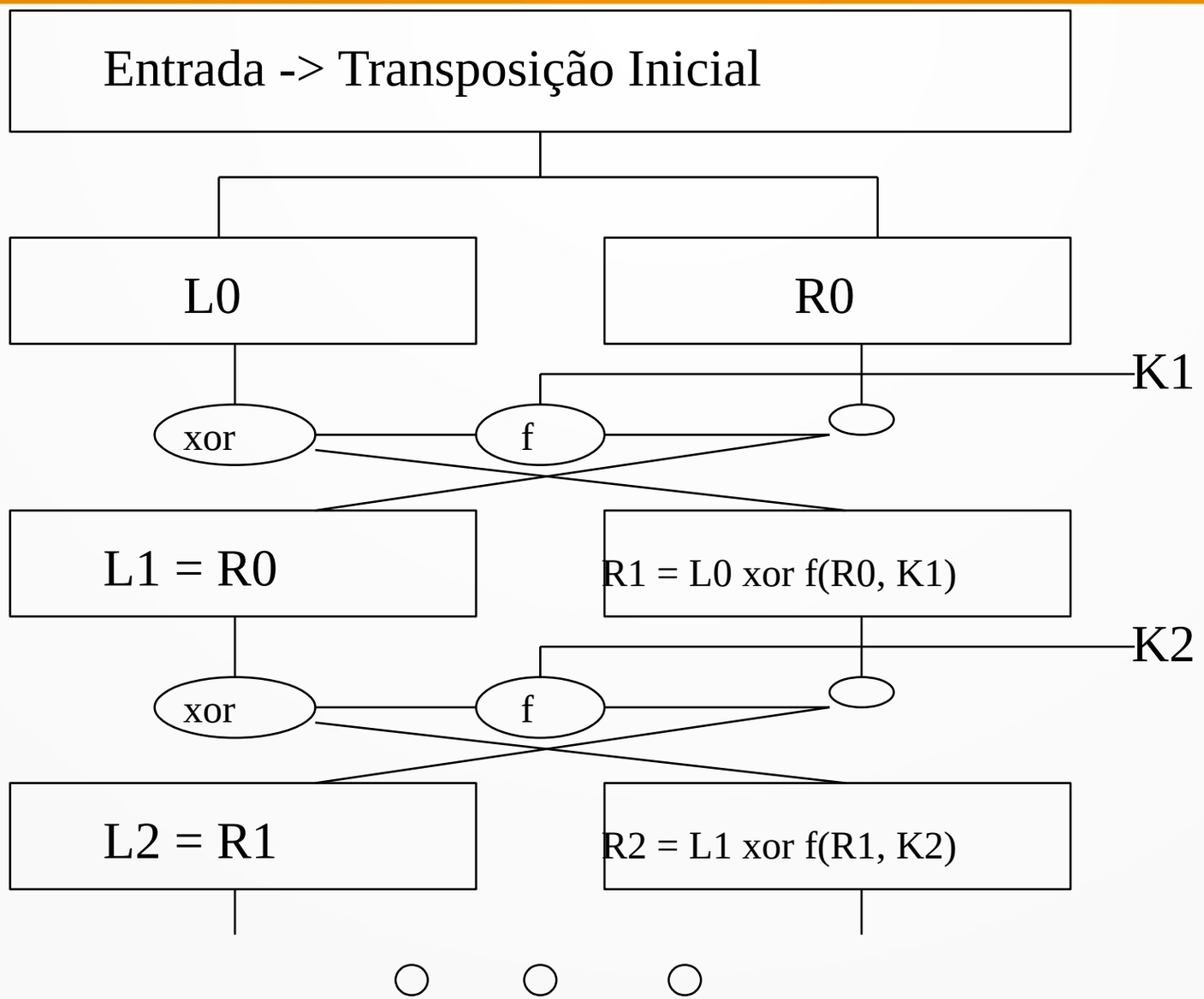
Por que bits de paridade?

- Permitem checar automaticamente se houve troca de um bit
- Mas... não é possível descobrir a troca de 2 bits!
- Se a chave tiver erros, o usuário descobriria ao usá-la
- O algoritmo fica 256 vezes menos seguro do que se as chaves fossem de 64 bits
- Nem chaves de 128 bits são realmente seguras para aplicações muito sensíveis hoje

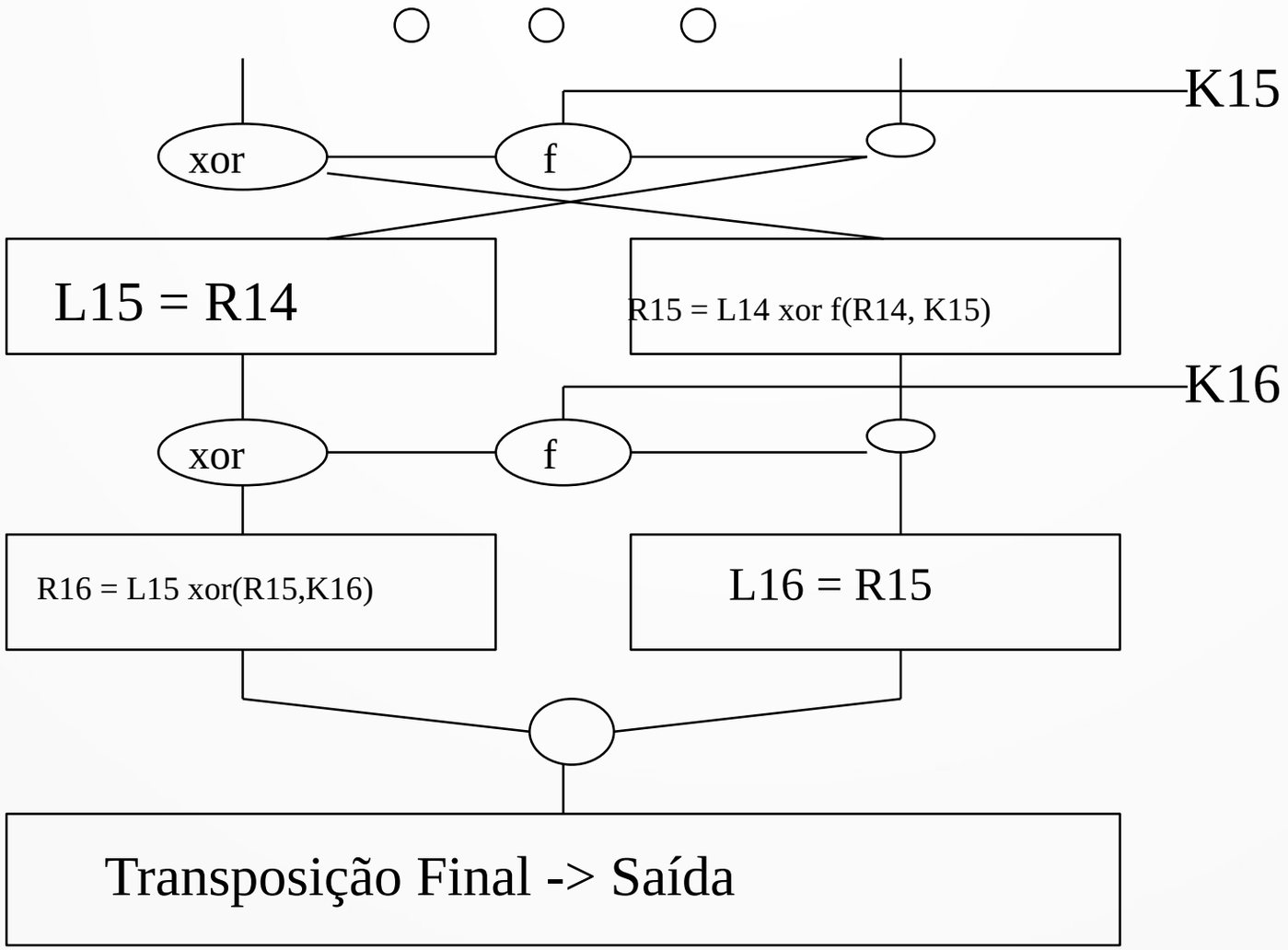
Como Funciona o DES?

- Inicialmente os 64 bits do bloco de entrada sofrem uma transposição
- Em seguida há 16 passos que fazem uma transformação nos dados
- Estes 16 passos são parametrizados por 16 chaves diferentes de 48 bits obtidas da chave de 56 bits
- No final, ocorre uma transposição inversa à inicial

O Algoritmo DES



O Algoritmo DES – cont.



DES: As Transposições

- Um transposição simples no início, outra no final
- Principal motivo de sua presença: dificultar implementação em software
- Não aumenta a segurança
- Nos 16 passos parametrizados, as chaves também são transpostas

DES: As Transposições

Transposição Inicial

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

Transposição Final

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

DES: Mais sobre as Transposições

- Os números das duas tabelas são a ordem inicial dos bits na entrada para a primeira transposição
- A transposição final inverte a inicial (ex. 1-40)
- Os bits do 1º byte são mapeados para as 8as posições dos bytes da saída, 2o -> 7a ....
- do iº byte para as (9-i)as posições
- Se forem ASCII: 8o bit de cada byte = zero o quinto byte da saída fica todo zerado!

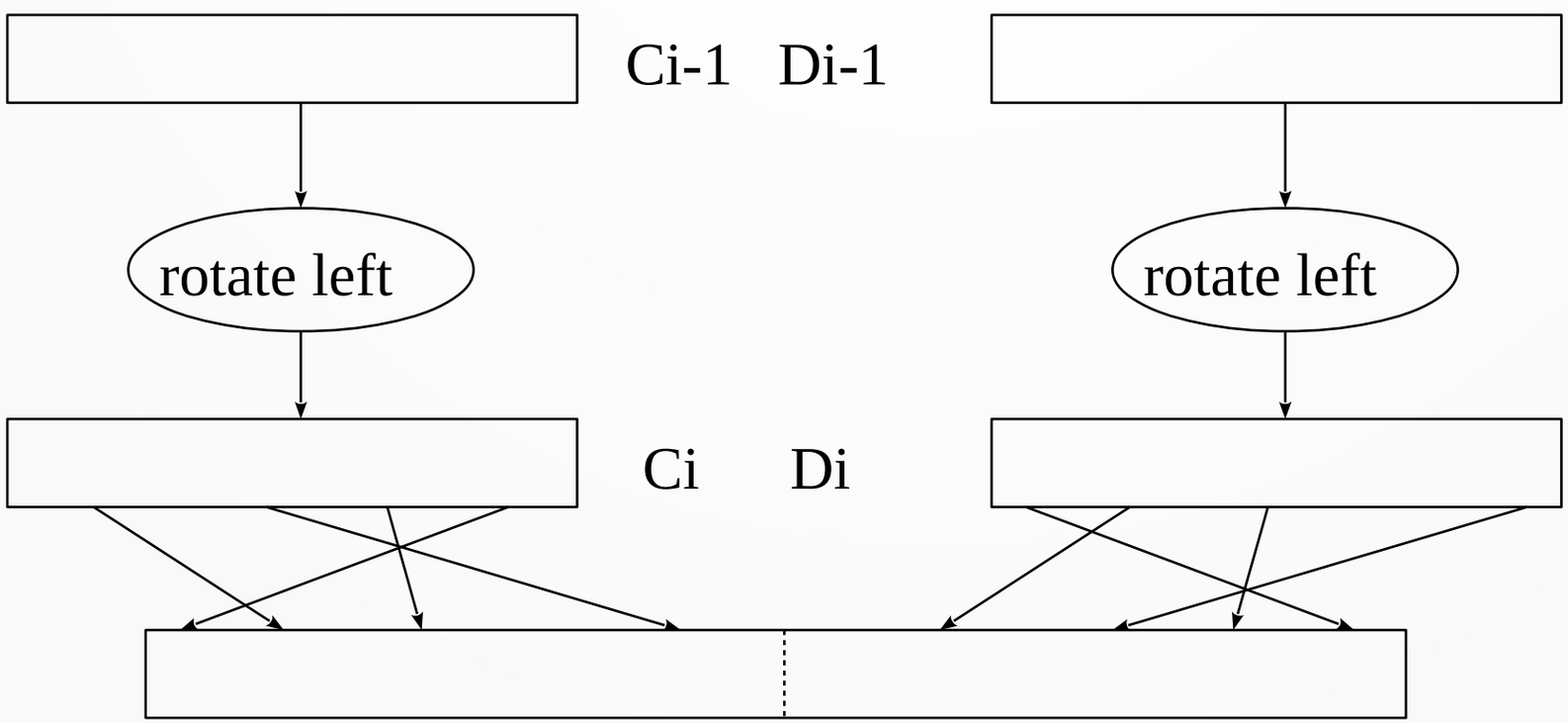
Uma Chave por Passo

- São 16 chaves diferentes de 48 bits cada
- Todas são obtidas a partir da chave original (secreta) que tem 64 bits (56 + 8 paridade)
- Inicialmente é feita uma transposição, cujo resultado é duas metades (28 bits) C0 e D0:

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

Para Gerar as Chaves

- Nas rodadas 1, 2, 9 e 16 rotaciona 1 bit, nas demais rodadas 2 bits são rotacionados:



As Chaves dos Passos: DES

- A transposição final de cada rodada é importante
- Bits 9, 18, 22 e 25 são descartados de Ci
- Bits 35, 38, 43 e 54 são descartados de Di

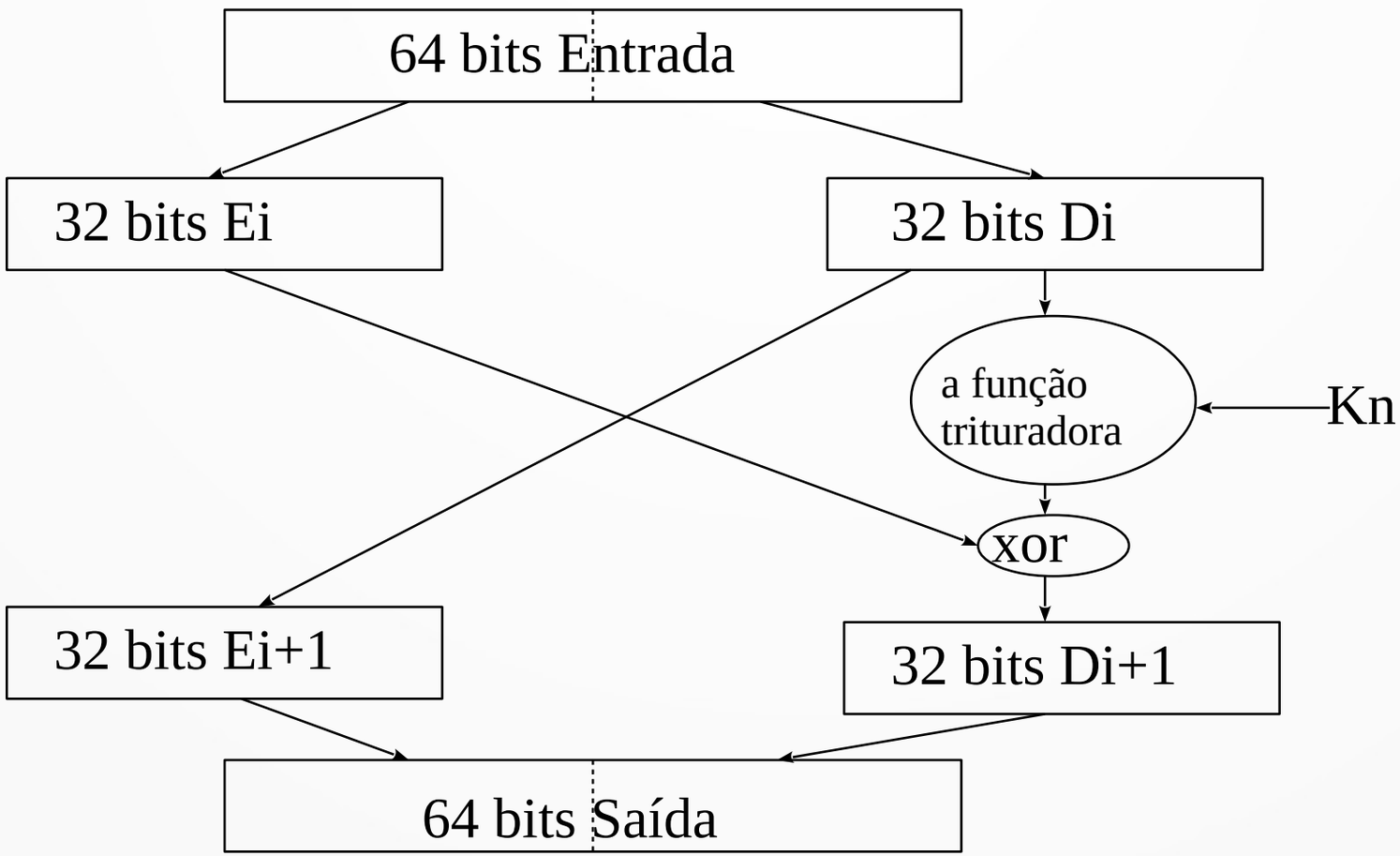
esquerda

14 17 11 24 1 5
3 28 15 6 21 10
23 19 12 4 26 8
16 7 27 20 13 2

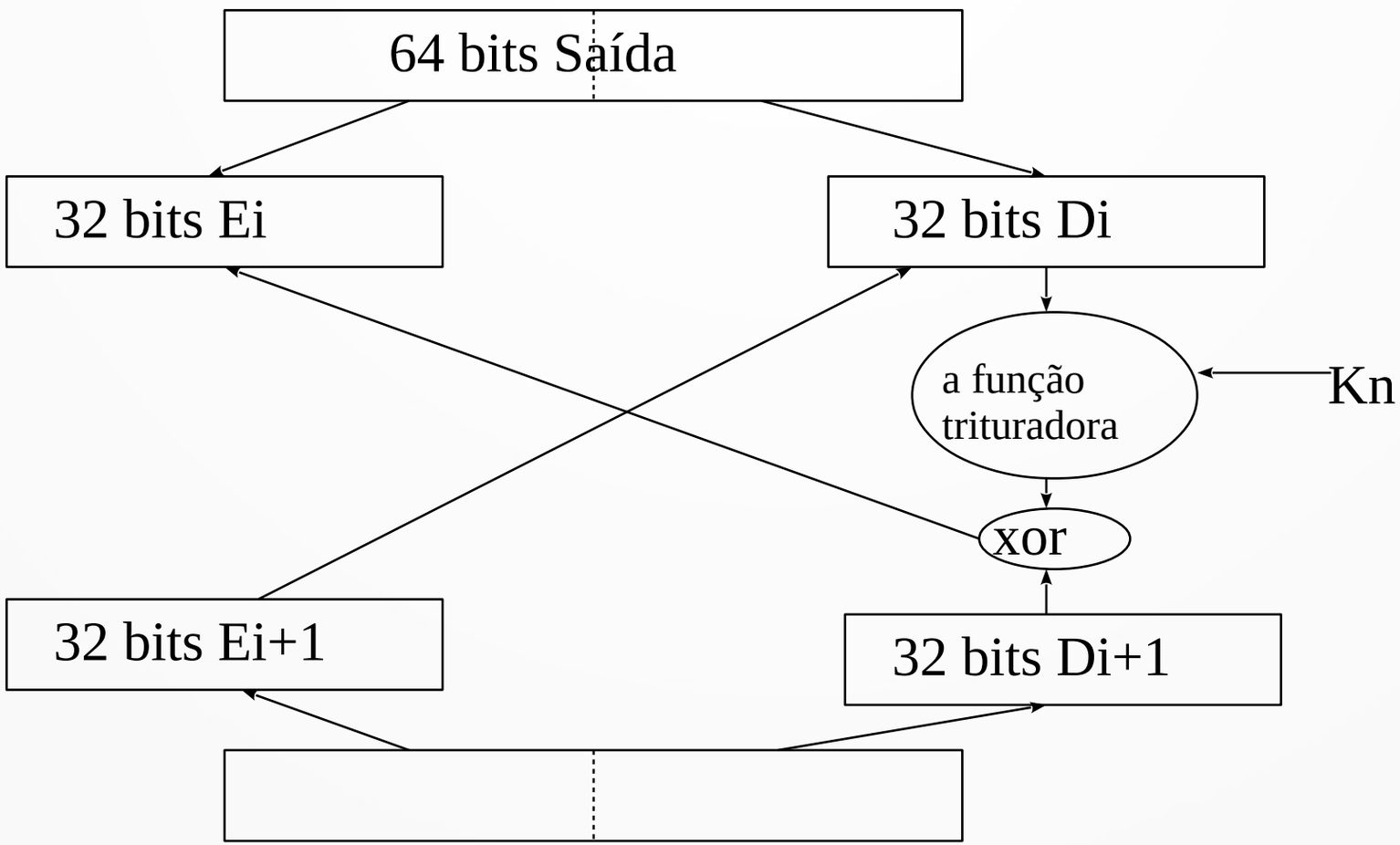
direita

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: 1 Rodada Criptografando



DES: 1 Rodada Descriptografando



A Função Trituradora

- Entrada: 32 bits (Di) + 48 bits chave (Ki)
- Saída: 32 bits XOR Ei -> Ri+1
- Inicialmente: expande os 32 bits p/ 48 bits
- grupos de 4 bits --> 6 bits
- bits adjacentes aos blocos são duplicados
- ex: 1111001111 --> 111100 & 001111
- Ocorre então um XOR com a chave
- Cada bloco de 6-bits entra numa S-box

As S-Boxes

- Cada S-box recebe 6 bits na entrada, e produz 4 bits na saída
- Como há 64 (26) entradas e 16 (24) saídas
- Várias entradas são mapeadas para cada saída
- As 8 S-boxes são totalmente especificadas por tabelas

Como Exemplo: Veja a S-Box3

- 4 bits mais internos, 2 bits das pontas:

	0000	0001	0010	....
00	1010	0000	0010	....
01	1101	0111	0000	....
10	1101	0110	0100	....
11	0001	1010	1101	....

A Função Trituradora – Cont.

- Os 8X4=32 bits que saem das S-boxes são então agrupados e ocorre uma transposição
- O mapeamento parece aleatório:

```
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: Chaves para Evitar

- A comunidade sugere que 16 chaves sejam evitadas
- As duas metades de 28 bits são um de:
- 0000, 1111, 0101, 1010
- Outras chaves também podem ser inseguras, por exemplo 000001 ou 000010

DEScriptografando

- Na verdade basta rodar o algoritmo de trás para frente
- A transposição final é feita primeiro, seguida do passo 16, passo 15, até o passo 1;
- Aí as metades são trocadas, e se faz a transposição inicial

A “Lógica” do DES

- “It is unfortunate that the design of DES was not more public. We don’t know why..”
- Se a S-box 3 for trocada pela S-box 7, foi mostrado que o algoritmo é uma ordem de magnitude mais fraco
- Os projetistas levaram em conta uma série de ataques ao tomar decisões sobre o algoritmo

AES: Substituiu o DES

- O DES deixou de ser forte o suficiente para aplicações sensíveis
- A agência do governo americano responsável pelos padrões de criptografia abriu um edital internacional
- Objetivo: evitar suspeitas que um algoritmo proposto pelo governo dos EUA tivesse uma “porta dos fundos”

AES: Os Requisitos

- O objetivo era um algoritmo de criptografia de chave secreta
- O projeto do algoritmo deveria ser publicado
- Chaves de 128, 192, and 256 bits deveriam ser permitidas
- O algoritmo deveria ser adequado para implementações em hardware e software

AES: Advanced Encryption Standard

- Em janeiro de 1997 o edital foi publicado
- 15 propostas foram recebidas
- Foram organizados workshops em que as propostas foram apresentadas e avaliadas
- Incentivando espírito crítico
- Dos 15, 5 foram selecionados e mais workshops organizados
- Em outubro do ano 2000 a proposta “Boneca do Reno” (Rijndael – *Rhine Doll)* dos pequisadores belgas Joan Daemen and Vincent Rijmende se transformou no AES

AES: Fatos

- Da mesma forma que DES, o Rijndael usa substituição e transposição em múltiplas rodadas
- O número de rodadas varia de 10 para chaves de 128 bits em blocos de 128 bits até 14 rodadas para o máximo
- Todas as operações executadas sobre bytes (octetos) para permitir implementações eficientes em hardware e software
- DES precisa acessar bits e isso torna implementações em software pouco eficientes DES
- O AES é um algoritmo muito veloz: um processador de 2-GHz atinge taxas de criptografia de 700 Mbps, permite criptograr até uma dúzia de videos 4K em tempo real
- Hoje implementado em diversos processadores

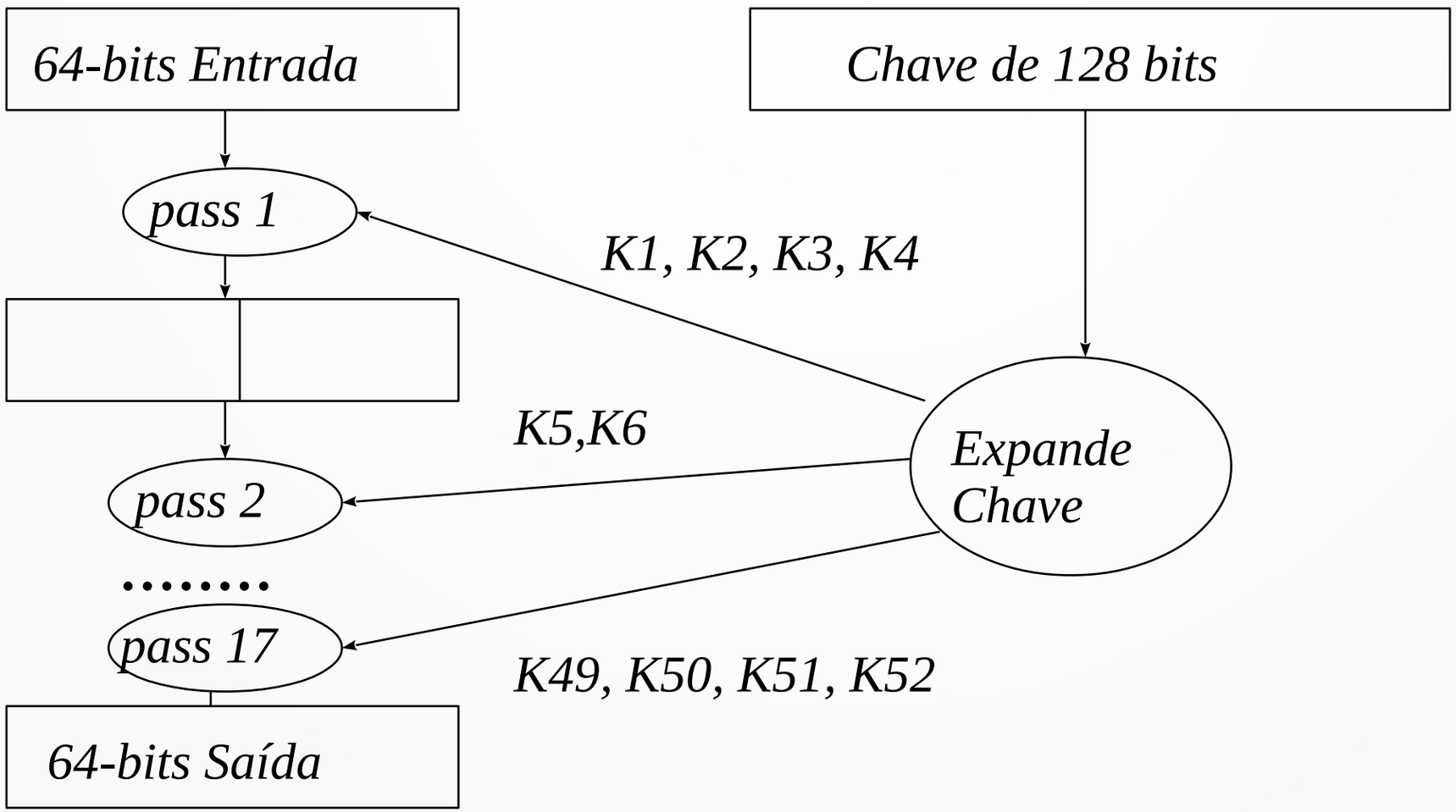
IDEA

- International Data Encryption Algorithm
- Autores acadêmicos (objetivo: acadêmico)
- Eficiente implementar em software
- Criptografa blocos de 64 bits -> 64 bits
- Chave de 128 bits
- Relativamente antigo (1991), nenhuma fragilidade foi encontrada

IDEA: O Algoritmo

- Opera em rodadas, como o DES
- A chave de 128 bits é usada para gerar 52 subchaves de 16 bits cada, e uma de 4 bits
- IDEA e DES têm uma função trituradora que não é reversível
- executada tb para descriptografar
- No IDEA, a relação das chaves para criptografar/descriptografar é diferente do DES (onde as chaves são as mesmas)

IDEA: Estrutura Básica



Rodadas do IDEA

- O algoritmo tem 17 passos agrupados em 8 rodadas, mais uma transformação final
- Em cada rodada, cada bit de saída depende de cada bit de entrada

Criptografia de Múltiplos Blocos

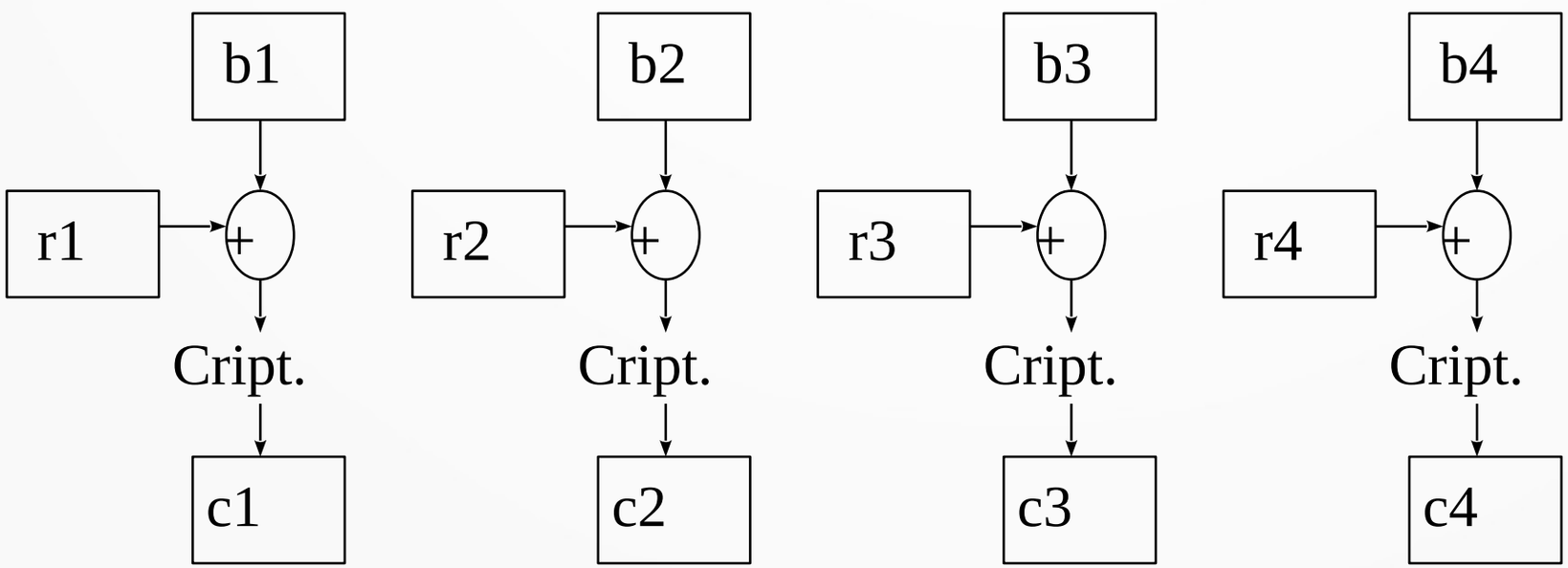
- Até agora:
- bloco 64 bits texto original na entrada
- bloco 64 bits texto criptografado na saída
- Como criptografar mais de 64 bits?
- Quatro esquemas padrão são definidos para o DES (aplicável IDEA):
- ECB, CBC, CFB, OFB

Modo ECB: Electronic Code Book

- O método mais simples: basta criptografar cada bloco de 64 bits do texto original sequencialmente & independentemente
- Já vimos os problemas que isso acarreta com o DES...
- troca de blocos de lugar não é detectável
- dois blocos idênticos têm saída idêntica -> informação para quebrar

Cipher Block Chaining: CBC

- Evitando que dois blocos iguais sejam criptografados da mesma maneira;
- Uma idéia (ainda não é CBC!):r1-n random

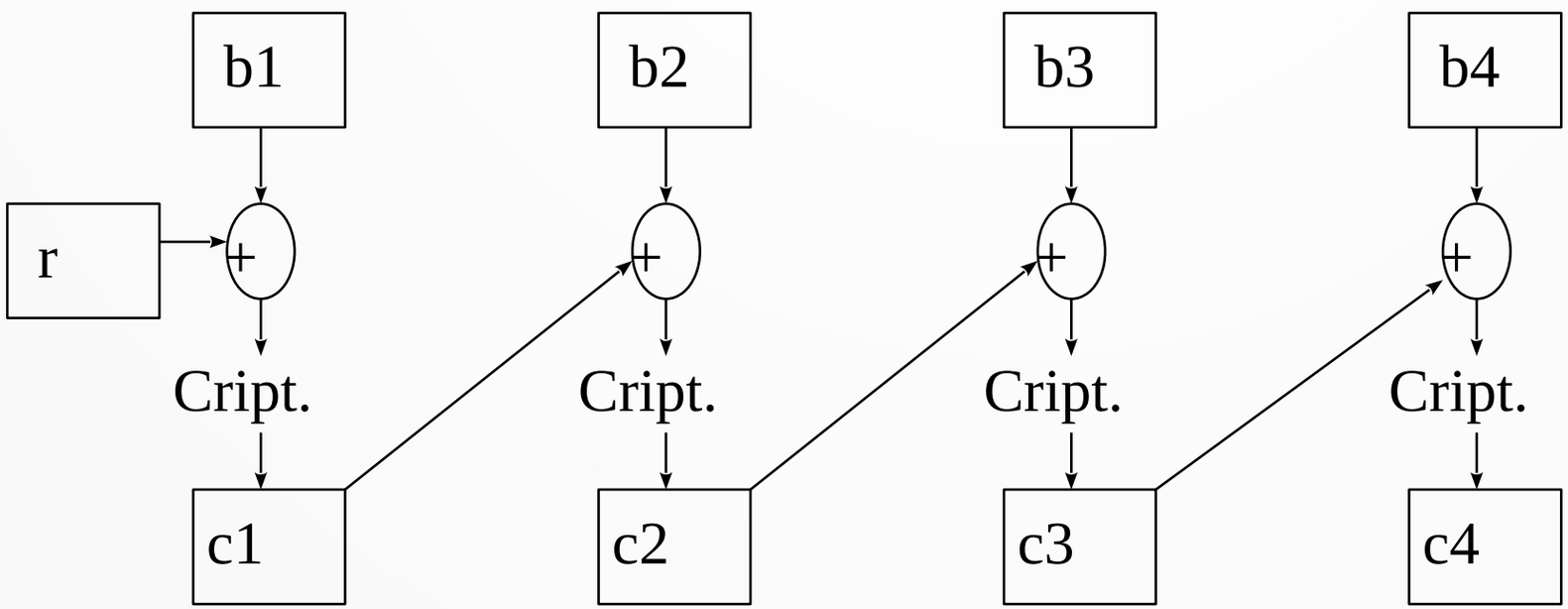


XOR com Números Aleatórios: Problemas

- Para cada bloco enviado, é necessário transmitir um número aleatório
- Ainda é possível rearranjar os blocos, isto é, trocá-los de lugar, sem que isso seja detectado

CBC – Cont.

- Uma solução mais eficiente: usar o bloco criptografado anteriormente no XOR:



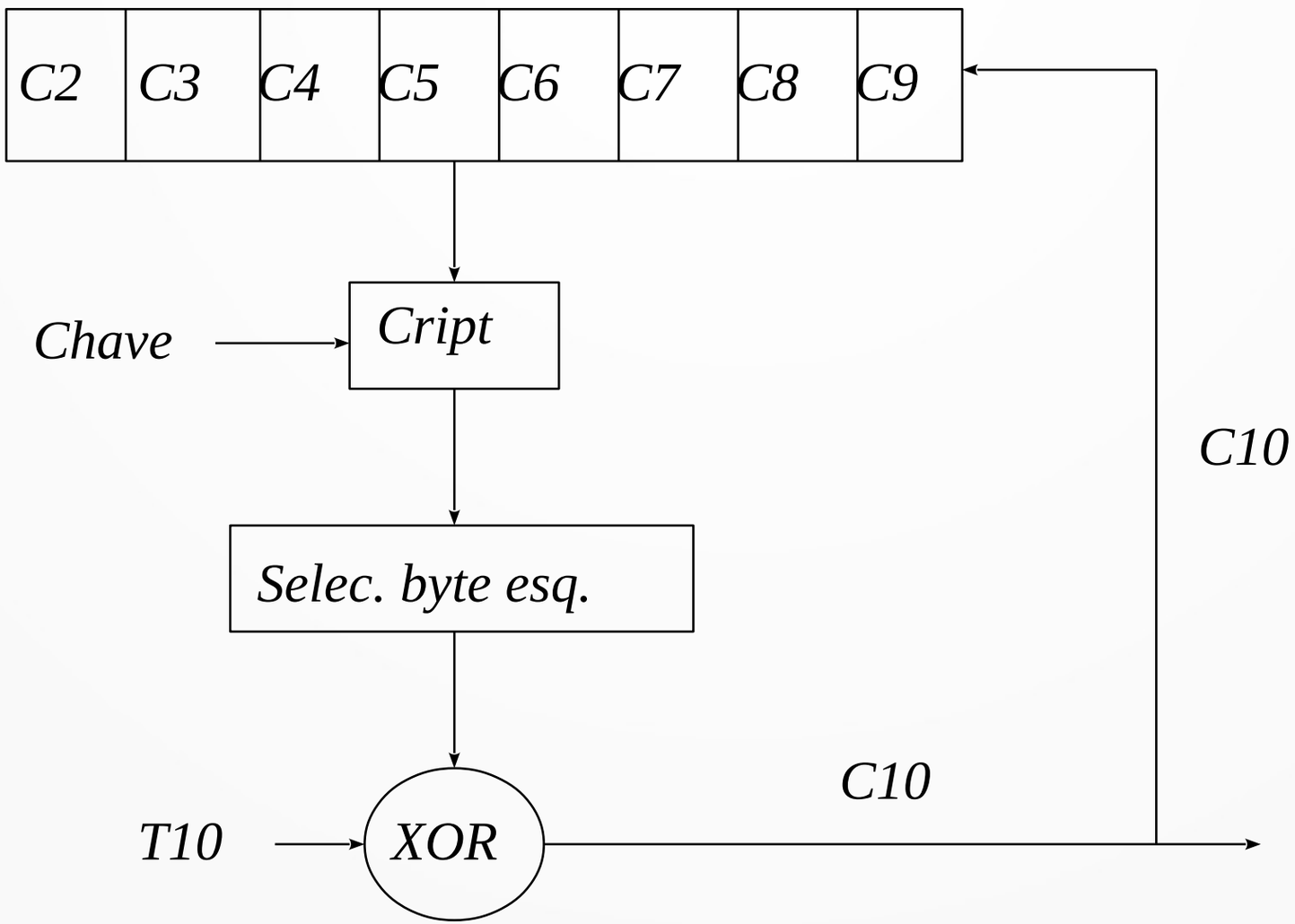
Cipher Block Chaining: Aplicação

- É necessário chegar o primeiro bloco de 64 bits para o emissor começar a descriptografar
- Não é adequado para aplicações interativas, com mensagens < 8 bytes
- Neste caso: Cipher Feedback Mode

Cipher Feedback Mode: CFB

- Os bytes são criptografados um a um
- Os últimos 8 bytes criptografados são guardados num registrador
- O bloco é então criptografado mais uma vez, e um byte é selecionado
- É feito um XOR deste byte com o próximo byte de dados a ser enviado

CFB: Byte a Byte



Output Feedback Mode

- No CFB quando um bit é trocado na transmissão: 64 bits de dados estão perdidos
- OFB: quando um bit é trocado, apenas 1 bit é perdido na saída
- Preço: é um método menos seguro

Outros Modos

- Além dos vistos: há muitas outras possibilidades
- É comum o uso de criptografia dupla ou tripla, usando o mesmo algoritmo
- “Triple encryption DES” é uma das formas mais populares de uso do DES quando se quer maior segurança e o mesmo algoritmo padrão

Conclusão

- Criptografia com chave secreta
- DES
- AES
- IDEA
- Criptografando múltiplos blocos: modos

Obrigado!

Lembrando: a página da disciplina é:
https://www.inf.ufpr.br/elias/topredes