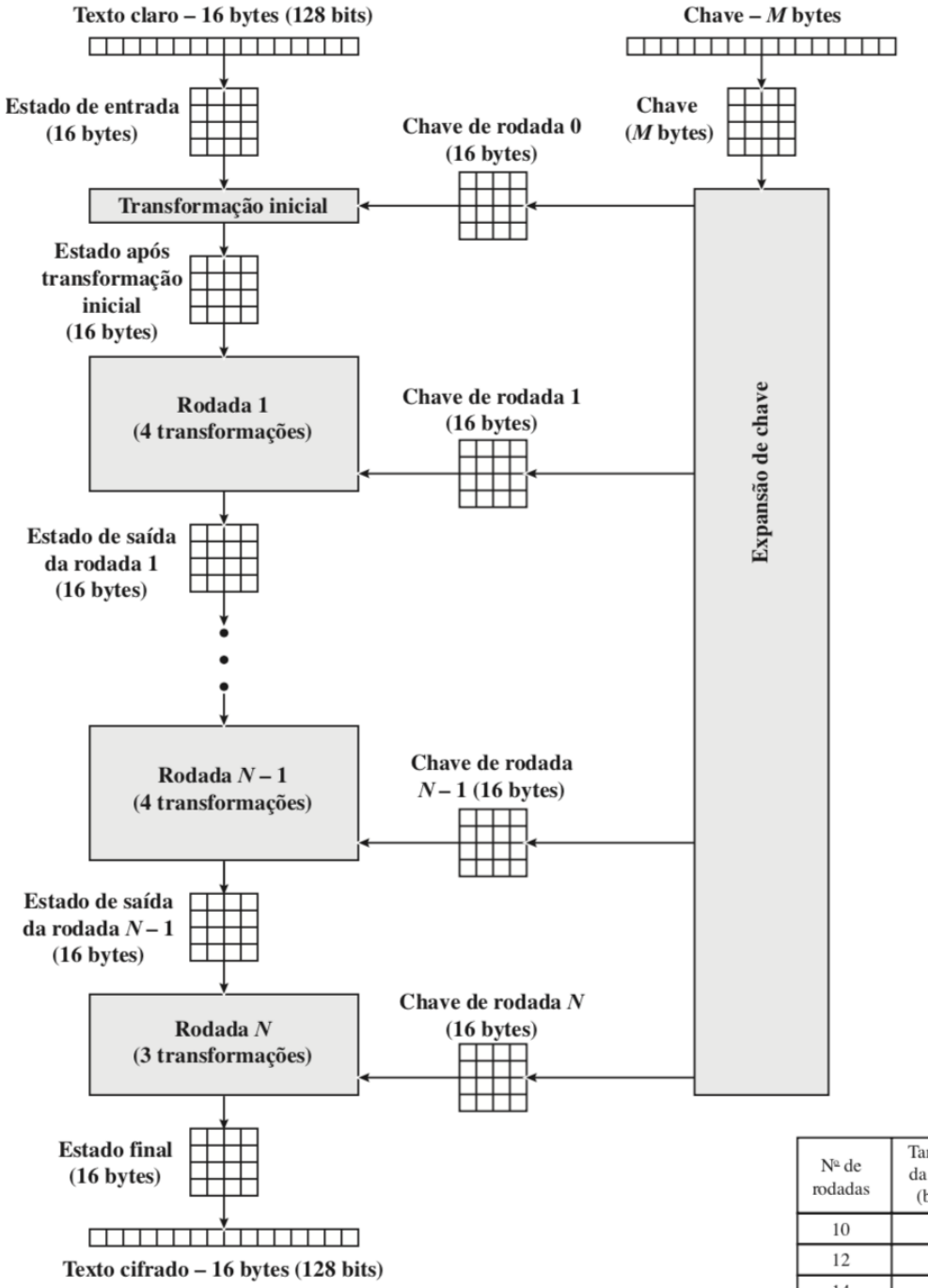
Segurança da Informação– GBC083

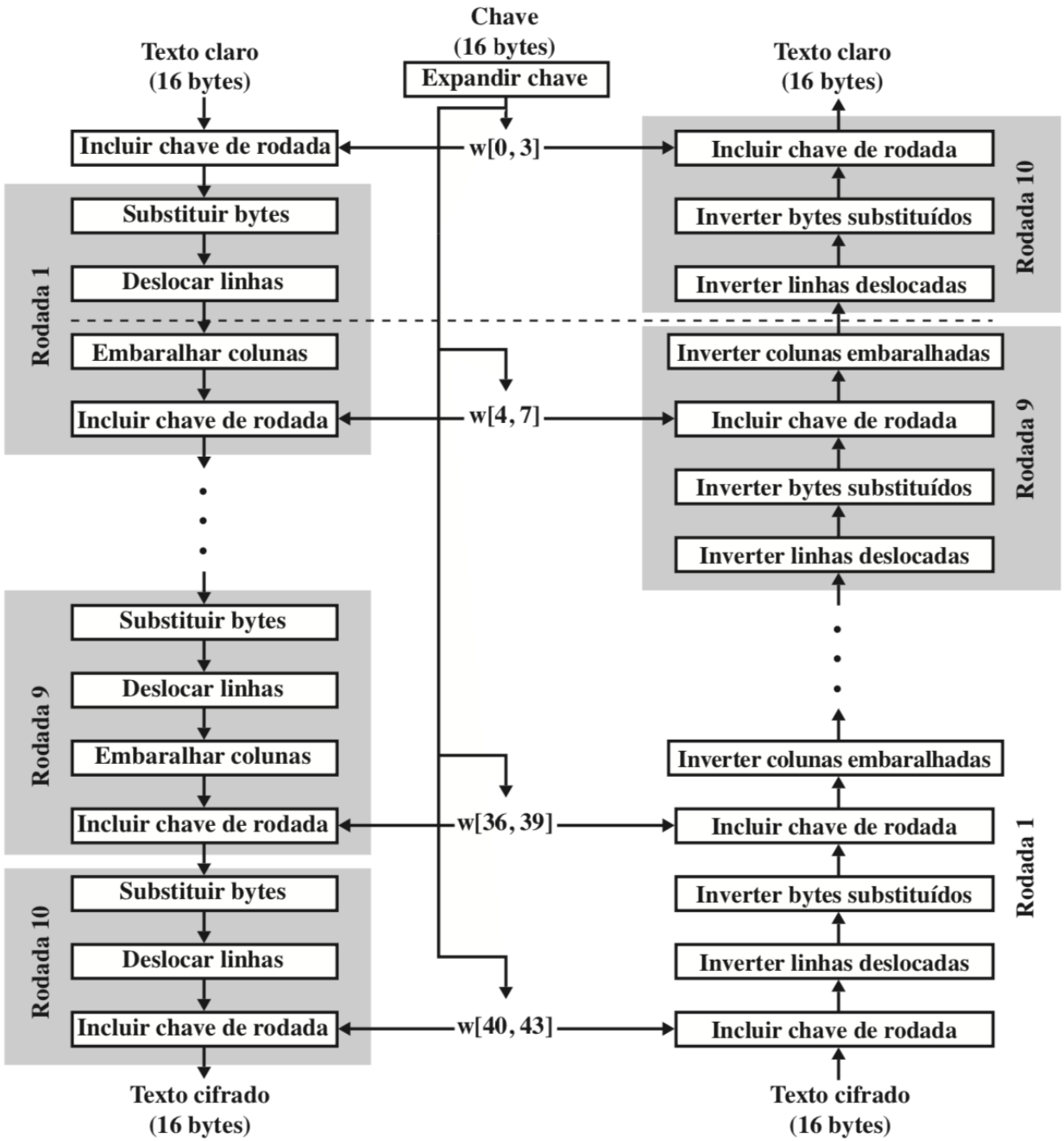
Prof. Rodrigo Sanches Miani – FACOM/UFU

Aula passada

Segurança da Informação– GBC083



Nº de rodadas	Tamanho da chave (bytes)
10	16
12	24
14	32


Chave
(16 bytes)
Expandir chave
Texto claro
(16 bytes)
Incluir chave de rodada
w[0, 3]
Substituir bytes
Deslocar linhas
Embaralhar colunas
Incluir chave de rodada
w[4, 7]
Rodada 1
Substituir bytes
Deslocar linhas
Embaralhar colunas
Incluir chave de rodada
w[36, 39]
Rodada 9
Substituir bytes
Deslocar linhas
Incluir chave de rodada
w[40, 43]
Rodada 10
Texto cifrado
(16 bytes)
Texto claro
(16 bytes)
Incluir chave de rodada
Inverter bytes substituídos
Inverter linhas deslocadas
Rodada 10
Inverter colunas embaralhadas
Incluir chave de rodada
Inverter bytes substituídos
Inverter linhas deslocadas
Rodada 9
Inverter colunas embaralhadas
Incluir chave de rodada
Inverter bytes substituídos
Inverter linhas deslocadas
Incluir chave de rodada
Rodada 1
Texto cifrado
(16 bytes)

Tópicos da aula

Segurança da Informação– GBC083

Tópicos da aula

- Modos de operação
- *Padding*
- ECB
- CBC

Modos de operação

Segurança da Informação– GBC083

Cifras de bloco

Uma cifra de bloco usa um bloco de texto de tamanho fixo com comprimento de *b* bits e uma chave como entrada, e produz um bloco de *b* bits de texto cifrado.

Cifras de bloco

Uma cifra de bloco usa um bloco de texto de tamanho fixo com comprimento de *b* bits e uma chave como entrada, e produz um bloco de *b* bits de texto cifrado.

- Pergunta: e se a quantidade de texto claro a ser cifrado tiver mais de *b* bits?

Cifras de bloco

Uma cifra de bloco usa um bloco de texto de tamanho fixo com comprimento de *b* bits e uma chave como entrada, e produz um bloco de *b* bits de texto cifrado.

- Pergunta: e se a quantidade de texto claro a ser cifrado tiver mais de *b* bits?
  - Resposta: dividir o texto claro em *n* blocos de *b* bits.

Cifras de bloco

- Ok... Iremos dividir o texto original em blocos de tamanho *b* bits.

- Como isso seria feito na prática? Ou seja, como o algoritmo de criptografia seria aplicado em tais blocos? Ideias?

Modo de cifras de bloco - proposta

Suponha que a mensagem em claro M é composta de três blocos b1, b2, b3. Dado uma chave K, eu poderia elaborar o seguinte esquema:

C1 = E(b1,K)

C2 = E(b2,K)

C3 = E(b3,K)

C = C1 || C2 || C3

Modo de cifras de bloco - proposta

- O modo anterior faz sentido?
  - Ele existe! É chamado de ECB. Veremos à seguir...

- Alguém consegue visualizar alguma limitação em tal modelo?

Modo de cifras de bloco - padronização

- Para empregar uma cifra de bloco em diversas aplicações, cinco *modos de operação* foram definidos pelo NIST (SP 800-38A);
  - Já existe uma atualização da SP 800-38ª (https://www.nist.gov/news-events/news/2023/04/decision-revise-nist-sp-800-38a-recommendation-block-cipher-modes-operation) – 28/04/2023

- Basicamente, um modo de operação é uma técnica para melhorar o efeito de um algoritmo criptográfico ou adaptar o algoritmo para uma aplicação.

Modo de cifras de bloco - padronização

- Os cinco modos abrangem praticamente todas as aplicações possíveis para as quais uma cifra de bloco poderia ser usada. Esses modos são utilizados com qualquer cifra de bloco simétrica, como DES e AES.
  - ECB
  - CBC
  - CFB
  - OFB
  - CTR

Modo de cifras de bloco*

Modo	Descrição	Aplicação típica
Electronic codebook (ECB)	Cada bloco de bits de texto claro é codificado independentemente usando a mesma chave.	■ Transmissão segura de valores isolados (por exemplo, uma chave de encriptação)
Cipher block chaining (CBC)	A entrada do algoritmo de encriptação é o XOR dos próximos 64 bits de texto claro e os 64 bits anteriores de texto cifrado.	■ Transmissão de uso geral orientada a bloco ■ Autenticação
Cipher feedback (CFB)	A entrada é processada s bits de cada vez. O texto cifrado anterior é usado como entrada para o algoritmo de encriptação a fim de produzir saída pseudoaleatória, que é aplicada a um XOR com o texto claro para criar a próxima unidade de texto cifrado.	■ Transmissão de uso geral orientada a fluxo ■ Autenticação
Output feedback (OFB)	Semelhante ao CFB, exceto que a entrada do algoritmo de encriptação é a saída DES anterior, e são usados blocos completos.	■ Transmissão orientada a fluxo por canal com ruído (por exemplo, comunicação por satélite)
Counter (CTR)	Cada bloco de texto claro é aplicado a um XOR com um contador encriptado. O contador é incrementado para cada bloco subsequente.	■ Transmissão orientada a bloco de uso geral ■ Útil para requisitos de alta velocidade

- * O termo "DES" na tabela refere-se a qualquer algoritmo simétrico de bloco.

Padding

Segurança da Informação– GBC083

Modo de cifras de bloco - proposta

Suponha que a mensagem em claro M é composta de três blocos b1, b2, b3. Dado uma chave K, eu poderia elaborar o seguinte esquema:

C1 = E(b1,K)

C2 = E(b2,K)

C3 = E(b3,K)

C = C1 || C2 || C3

Preenchimento de bits

O que aconteceria se o último bloco, b3, tiver um número de bits menor do que o necessário?

Preenchimento de bits

O que aconteceria se o último bloco, b3, tiver um número de bits menor do que o necessário?

Algum preenchimento deve ser feito! Como?

Padding

- O método usado para preencher bits em criptografia é conhecimento como *padding*;

- Existem diversas formas de se fazer isso, inclusive padrões amplamente consolidados no mercado como o PKCS#5 e o PKCS#7
  - Public-Key Cryptography Standards

Padding

Clear text consists of the following18 bytes:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB
```

In order to make this a multiple of 16 bytes (the AES block size), we must add 14 bytes. Each byte will contain the value 0x0E, which is 14, the total number of padding bytes added. The result is that the padded clear text is as follows:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB0E0E0E0E0E0E0E
0E0E0E0E0E0E0E0E
```

https://www.ibm.com/docs/en/zos/2.4.0?topic=rules-pkcs-padding-method

Padding

Após a decifragem, como o sistema sabe onde começa o texto claro?

Clear text consists of the following18 bytes:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB
```

In order to make this a multiple of 16 bytes (the AES block size), we must add 14 bytes. Each byte will contain the value 0x0E, which is 14, the total number of padding bytes added. The result is that the padded clear text is as follows:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB0E0E0E0E0E0E0E
0E0E0E0E0E0E0E0E
```

https://www.ibm.com/docs/en/zos/2.4.0?topic=rules-pkcs-padding-method

Padding

Após a decifragem, como o sistema sabe onde começa o texto claro?

Como sempre vai ter *padding*, o sistema lê o último byte da cadeia toda e vê que foram adicionados 14 bytes (0x0E). O sistema então conta, do fim para o início, 14 bytes e faz o devido "corte" no padding.

Clear text consists of the following18 bytes:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB
```

In order to make this a multiple of 16 bytes (the AES block size), we must add 14 bytes. Each byte will contain the value 0x0E, which is 14, the total number of padding bytes added. The result is that the padded clear text is as follows:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB0E0E0E0E0E0E0E
0E0E0E0E0E0E0E0E
```

- https://www.ibm.com/docs/en/zos/2.4.0?topic=rules-pkcs-padding-method

Padding

E se o texto claro já é um múltiplo de 16 bytes?

Padding de novo! Sempre tem *padding* se o sistema usa o PKCS. Qual o impacto disso?

Clear text consists of the following18 bytes:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB
```

In order to make this a multiple of 16 bytes (the AES block size), we must add 14 bytes. Each byte will contain the value 0x0E, which is 14, the total number of padding bytes added. The result is that the padded clear text is as follows:

```
F14ADBDA019D6DB7 EFD91546E3FF8444 9BCB0E0E0E0E0E0E0E
0E0E0E0E0E0E0E0E
```

- https://www.ibm.com/docs/en/zos/2.4.0?topic=rules-pkcs-padding-method

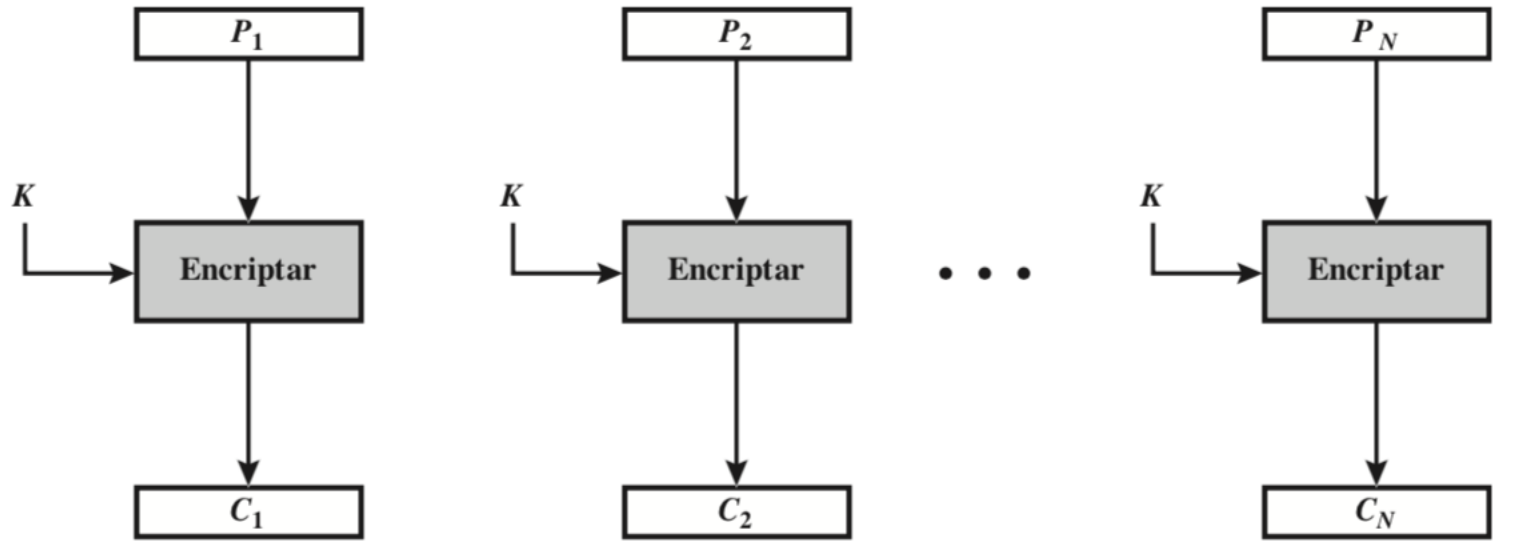
ECB – Electronic Codebook

Segurança da Informação– GBC083

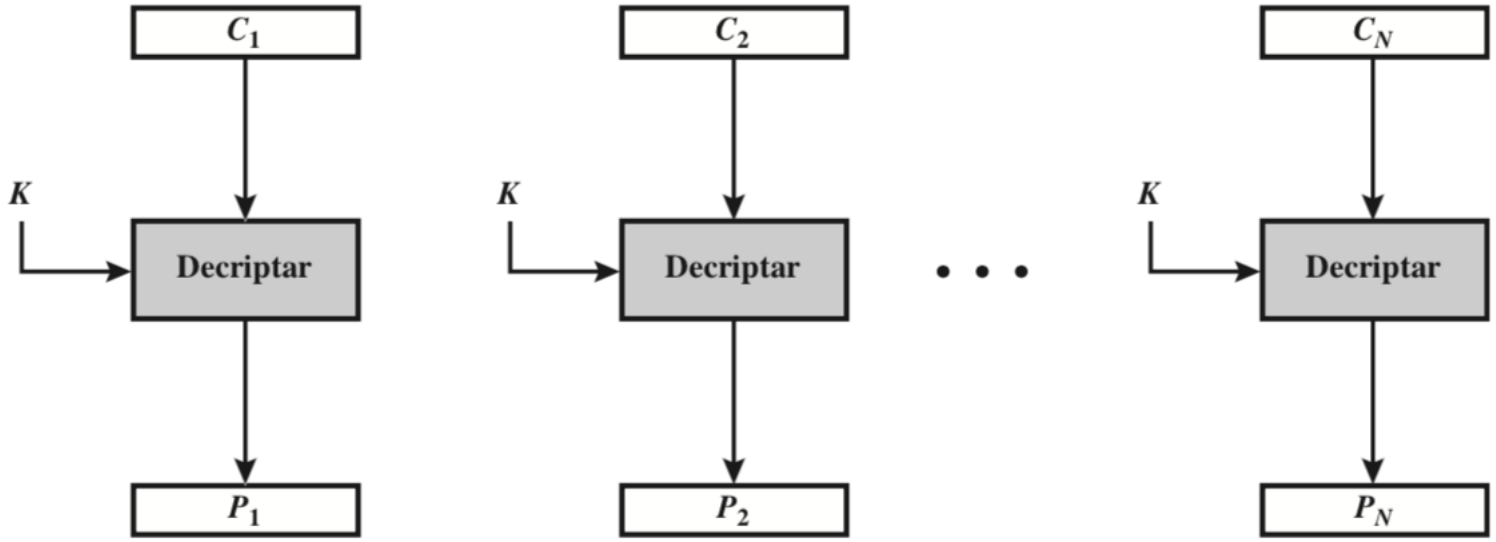
ECB

- Modo mais simples de operação de uma cifra de bloco;
- Texto claro é tratado por um bloco de cada vez;
- Cada bloco de texto claro é cifrado usando a mesma chave.

ECB - cifrar



ECB - decifrar



ECB - discussão

O que aconteceria se três blocos de texto claro fossem iguais, ou seja, P1=P3=P5?

ECB - discussão

O que aconteceria se três blocos de texto claro fossem iguais, ou seja, P1=P3=P5?

Os três blocos cifrados também seriam iguais!

ECB - discussão

- Essa característica do ECB o torna ideal para uma pequena quantidade de dados de entrada.
  - Número pequeno de blocos!

- ECB, definitivamente, não é recomendado para mensagens mais longas (grande quantidade de dados de entrada).
  - Tais tipos de mensagens podem ser estruturadas e fornecer pistas para o criptoanalista.

CBC – Cipher Block Chaining

Segurança da Informação– GBC083

Limitações do ECB

- Como resolver as limitações do ECB?

Limitações do EBC

- Como resolver as limitações do ECB?
- Propor um novo modo de operação!
- Qual seria o objetivo de tal modo de operação?

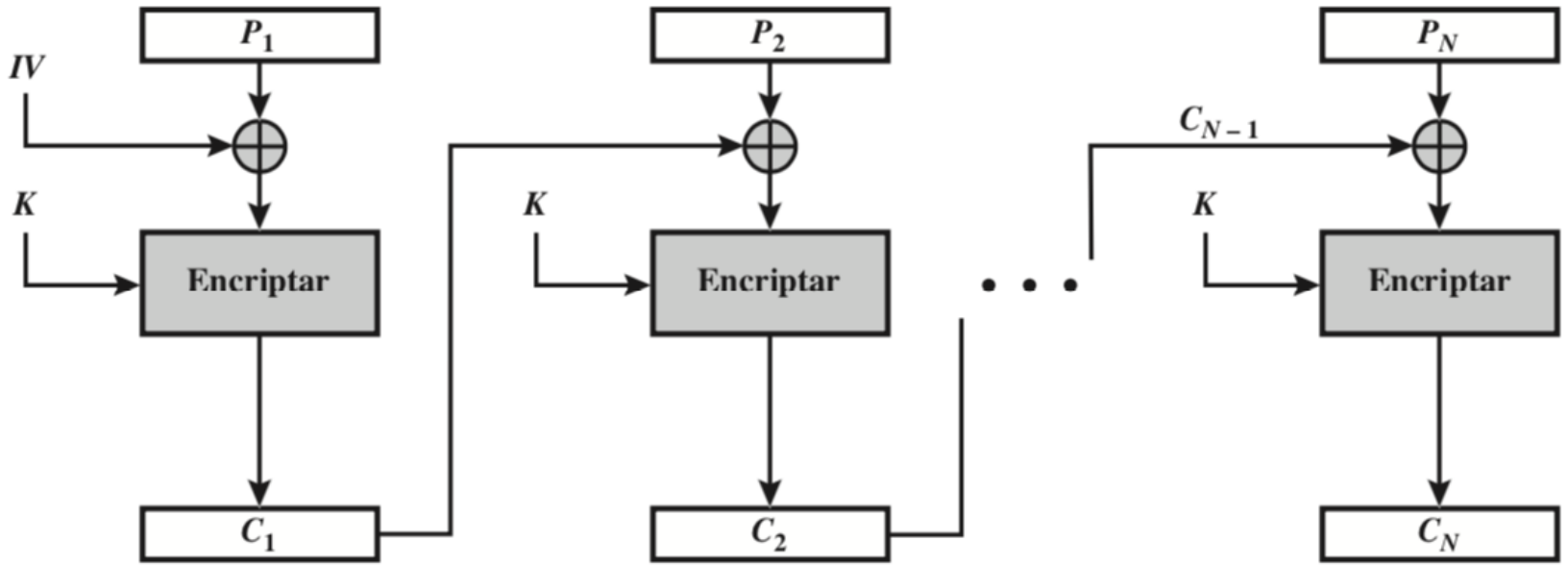
Limitações do EBC

- Como resolver as limitações do ECB?
- Propor um novo modo de operação!
- Qual seria o objetivo de tal modo de operação?
  - Mesmo bloco de texto claro, quando repetido, irá produzir textos cifrados diferentes.

Limitações do EBC

- Como resolver as limitações do ECB?
- Propor um novo modo de operação!
- Qual seria o objetivo de tal modo de operação?
  - Mesmo bloco de texto claro, quando repetido, irá produzir textos cifrados diferentes.
  - Como fazer isso?

CBC - cifrar



CBC - cifrar

- O que está acontecendo agora?
- Na primeira interação (bloco 1) o bloco de texto claro passa por um XOR com um tal de IV e só depois é cifrado;
- O texto cifrado é então usado como entrada na próxima interação (bloco 2) à partir de um XOR com o bloco 2 e assim sucessivamente...

CBC - cifrar

- A entrada da função de cifrar para cada bloco de texto claro não possui qualquer relacionamento **fixo** com o de texto claro;

- Portanto, padrões repetitivos de *b* bits não são expostos.

CBC – cifrar (IV)

- IV é chamado de vetor de inicialização (*initialization vector*). É um número que será usado na primeira iteração da cifragem;

Perguntas:

1. Qual a importância do IV?
2. Como gerar tal número?
3. Qual é o tamanho de tal número?
4. Quais cuidados devemos ter na geração de tal número?
5. Ambos (emissor e receptor) devem conhecê-lo?

CBC – cifrar (IV)

Perguntas:

1. Qual a importância do IV?

- O IV permite que o texto claro seja modificado na primeira rodada da execução do algoritmo.
- Nas rodadas subsequentes isso será feito com o bloco de texto cifrado anterior...

CBC – cifrar (IV)

Perguntas:

2. Como gerar tal número?

- A norma SP800-38A recomenda dois métodos:
  - Aplicar a função de criptografia usando a mesma chave em um **nonce**. O **nonce** precisa ser um bloco de dados exclusivo a cada execução da operação de criptografia. Por exemplo, pode ser um contador, um timestamp ou um número de mensagem.
  - Gerar um bloco de dados aleatório usando um gerador de números aleatórios.

CBC – cifrar (IV)

Perguntas:

3. Qual é o tamanho de tal número?

- Bloco de mesmo tamanho do bloco de texto claro (mesmo tamanho da cifra).

CBC – cifrar (IV)

Perguntas:

4. Quais cuidados devemos ter na geração de tal número?

- IV deve ser imprevisível para um terceiro. Em particular, para qualquer texto claro dado, não deverá ser possível prever o IV que estará associado ao texto claro antes da geração dele.
- Existem diversos ataques com base no conhecimento do IV. Isso deve ser evitado!

CBC – cifrar (IV)

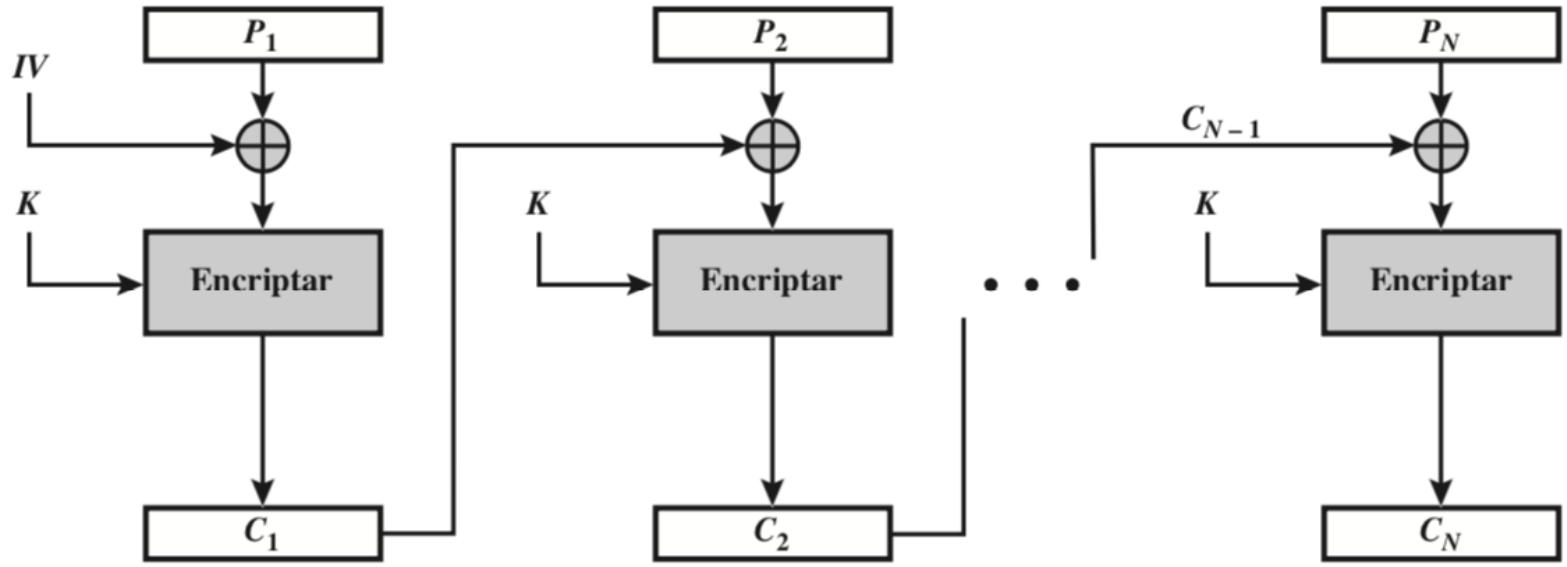
Perguntas:

5. Ambos (emissor e receptor) devem conhecê-lo?

- Sim! Isso está associado a maneira com que a operação decifrar do CBC é feita. Veremos a seguir.

CBC – decifrar

Dado que a cifragem é feita usando o seguinte modelo, como a decifragem deve ser feita?



CBC – decifrar

- A ideia é analisar a ordem da cifragem e tentar usar uma ordem inversa na decifragem...

CBC – decifrar

- A ideia é checar a ordem da cifragem e tentar usar uma ordem inversa na decifragem...

- Cifrar:
  1. P1+IV
  2. E(K,P1+IV)

CBC – decifrar

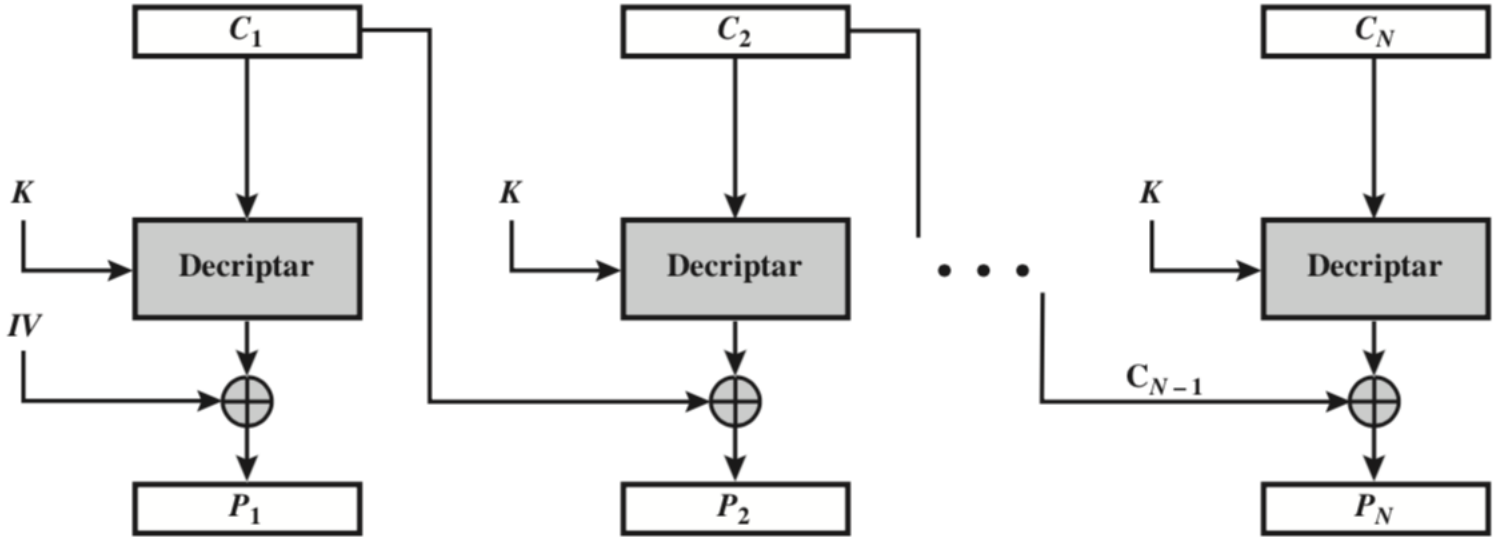
- A ideia é checar a ordem da cifragem e tentar usar uma ordem inversa na decifragem...

- Cifrar:
  1. P1+IV
  2. E(K,P1+IV)

- Usando essa lógica...

- Decifrar:
  1. D(K, C1)
  2. D(K,C1) + IV
  3. Ou seja, primeiro decifro o bloco e depois faço um XOR com o vetor de inicialização.
  4. Demonstre isso!

CBC – decifrar



Roteiro de estudos

1. Leitura das seções 6.2, 6.3, 6.4, 6.5 e 6.6 do livro "Criptografia e segurança de redes. Princípios e práticas". William Stallings;
2. Estudo das vídeo-aulas referentes ao tópico 8;
3. Resolução dos TP-3 e TP-4.