

# Cifras de bloco

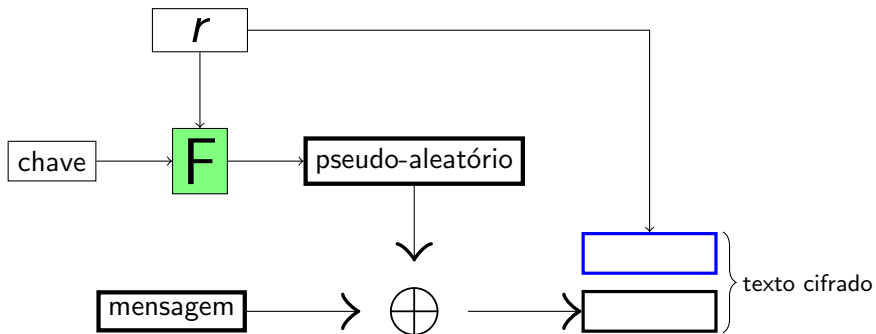
- ▶ Cifras de bloco são construções práticas de permutações pseudo-aleatórias
- ▶  $F : \{0, 1\}^n \times \{0, 1\}^m \rightarrow \{0, 1\}^m$ 
  - ▶  $n$  é comprimento da chave
  - ▶  $m$  é o comprimento do bloco
- ▶ Difícil diferenciar  $F_k$  de uniforme  $f \in Perm_m$  mesmo para atacantes em tempo  $\approx 2^n$

# AES (2002)

- ▶ Advanced Encryption Standard (AES) ou Rijndael
  - ▶ Padronizado por NIST em 2000 baseado em uma competição pública mundial que durou 3 anos
  - ▶ Tamanho do bloco = 128 bits
  - ▶ Tamanho da chave = 128, 192 ou 256 bits
- ▶ Padrão mais usado atualmente

# Encriptação segura a CPA (Ataques de texto em claro)

- ▶ Seja  $F$  uma função chaveada
- ▶  $Gen(1^n)$ : escolher uma chave uniforme  $k \in \{0, 1\}^n$
- ▶  $Enc_k(m)$  para  $|m| = |k|$ :
  - ▶ Escolher uniforme  $r \in \{0, 1\}^n$
  - ▶ Produzir texto cifrado  $\langle r, F_k(r) \oplus m \rangle$
- ▶  $Dec_k(\langle c_1, c_2 \rangle)$ : produzir  $c_2 \oplus F_k(c_1)$



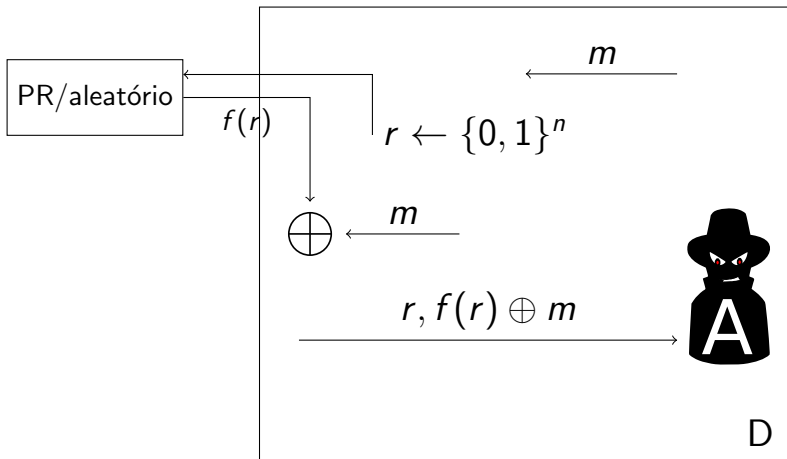
# Nota

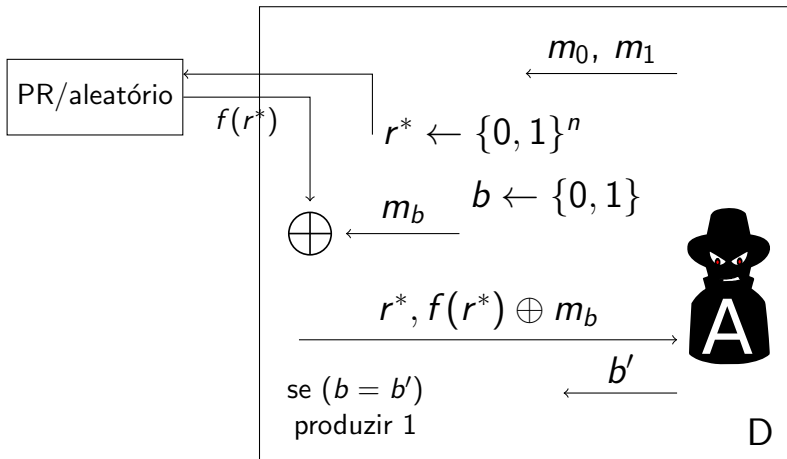
- ▶ A chave é tão longa quanto a mensagem
- ▶ mas esse esquema pode ser usado para encriptar múltiplas mensagens

# Segurança

- ▶ Teorema: se  $F$  é uma função pseudo-aleatória então este esquema  $\Pi$  é seguro para ataques CPA
- ▶ Prova por redução

- ▶ 1) o atacante  $D$  simula um oráculo de encriptação de uma vítima
- ▶ 2) o atacante  $A$  faz quantas encriptações forem necessárias







# Análise

- ▶ Seja  $\mu(n) = P(\text{PrivCPA}_{Adv, \Pi}(n) = 1)$
- ▶ Seja  $q(n)$  um limitando para o número de consultas de encriptação feio pelo atacante
- ▶ Se  $f = F_k$  para  $k$  uniforme, então a visão de  $Adv$  é exatamente como em  $\text{PrivCPA}_{Adv, \Pi}(n)$

$$\Rightarrow P_{k \leftarrow \{0,1\}^n}(D^{F_k(\cdot)} = 1) = P(\text{PrivCPA}_{Adv, \Pi}(n) = 1) = \mu(n)$$

# Análise

- ▶ Se  $f$  é uniforme, existem dois sub-casos
  - ▶  $r^*$  foi consultado em algum outro momento (evento **Repetir**)
  - ▶  $r^*$  não foi consultado em algum momento

$$\begin{aligned}P_f(D^{f(\cdot)} = 1) &= P_f(D^{f(\cdot)} = 1 \text{ e } \neg\mathbf{Repetir}) \\ &\quad + P_f(D^{f(\cdot)} = 1 \text{ e } \mathbf{Repetir}) \\ &\leq P_f(D^{f(\cdot)} = 1 | \neg\mathbf{Repetir}) + P(\mathbf{Repetir})\end{aligned}$$

- ▶  $P(\mathbf{Repetir}) \leq q(n)/2^n$
- ▶  $P_f(D^{f(\cdot)} = 1 | \neg\mathbf{Repetir}) = 0.5$

# Análise

- ▶ Como, de acordo com nossa premissa,  $F$  é pseudo-aleatório
  - ▶  $\Rightarrow |\mu(n) - P_f(D^{f(\cdot)} = 1)| \leq \epsilon(n)$
  - ▶  $\Rightarrow \mu(n) \leq P_f(D^{f(\cdot)} = 1) + \epsilon(n)$
  - ▶  $\leq 0.5 + q(n)/2^n + \epsilon(n)$
- ▶ Como, para qualquer polinômio  $q$ , o termo  $q(n)/2^n$  é negligível

$$P(\text{PrivCPA}_{\text{Adv}, \Pi}(n) = 1) = \mu(n) \leq 0.5 + \epsilon'(n)$$

# Resumo de encriptação segura contra CPA

- ▶ CPA modela atacante capaz de escolher textos em claros para encriptação
- ▶  $F_k(r)$  é uma função/permutação pseudo-aleatória chaveada que recebe e produz  $n = |m|$  bits
- ▶  $Enc_k(m) = (r, F_k(r) \oplus m)$ ,  $Dec_k(c_1, c_2) = F_k(c_1) \oplus c_2$
- ▶ Dessa forma, ao transmitir  $r$ , é possível usar a mesma chave  $k$  para cada bloco de mensagem  $m_1, m_2, m_3 \dots$

# Desvantagem

- ▶ O texto cifrado tem o dobro do comprimento ( $|r| + |F_k(r) \oplus m|$ ) da mensagem original
  - ▶ Diz-se que texto cifrado tem **expansão** de um fator de 2
- ▶ Podemos fazer melhor?
- ▶ Modos de operação: mais eficientes em termos de tamanho de mensagem encriptada

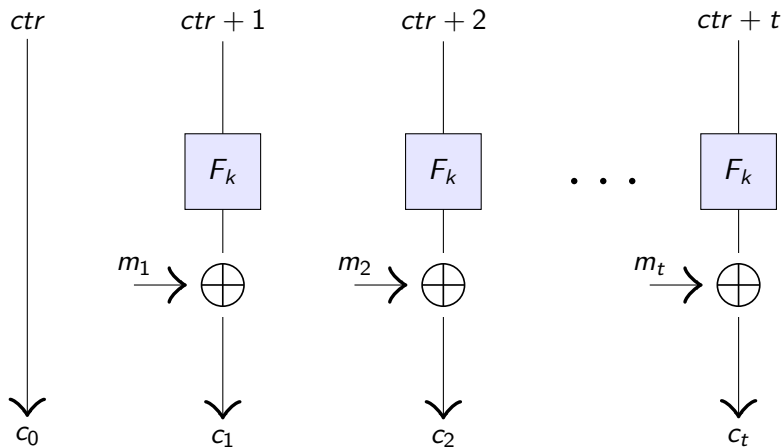
# Modos de operação

- ▶ Cifras podem atuar em sequências (streams) ou em blocos
- ▶ Cifras de streams
  - ▶ Sincronizados
  - ▶ Não sincronizados
- ▶ Cifras de blocos
  - ▶ Counter mode - CTR
  - ▶ Cipher Block Chaining - CBC
  - ▶ Eletronic Codebook - ECB

# Modo CTR (*counter*)

- ▶  $Enc_k(m_1, \dots, m_t)$ 
  - ▶ Escolher  $ctr \leftarrow \{0, 1\}^n$ , fixado  $c_0 = ctr$
  - ▶ Para  $i = 1 \dots t$  fazer:
    - ▶  $c_i \leftarrow m_i \oplus F_k(ctr + i)$
    - ▶ Produzir  $c_0, c_1, \dots, c_t$
- ▶ No modo CTR, A expansão do texto cifrado é de somente 1 bloco

# Modo CTR – Counter





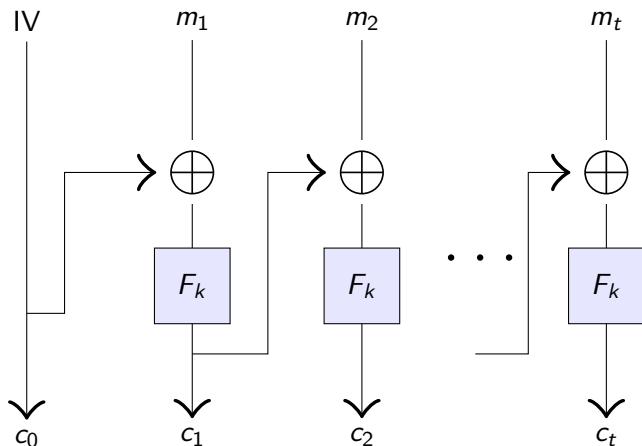
# Modo CTR

- ▶ Teorema: Se  $F$  é uma função pseudo-aleatória
- ▶ Ideia da prova:
  - ▶ Por simplicidade, assumir que todas as mensagens tem comprimento  $t$
  - ▶ A sequência  $F_k(ctr_i + 1), \dots, F_k(ctr_i + t)$  usada para encriptar a  $i$ -ésima mensagem é pseudo-aleatória
    - ▶ E também é independente entre sequências a não ser que  $ctr_i + j = ctr_{i'} + j'$  para algum  $i, j, i', j'$

# Modo CBC

- ▶  $Enc_k(m_1, \dots, m_t)$ 
  - ▶ Escolher aleatoriamente  $c_0 \leftarrow \{0, 1\}^n$  chamado de IV
  - ▶ IV = initialization vector
  - ▶ Para  $i \in 1 \dots t$ :
    - ▶  $c_i = F_k(m_i \oplus c_{i-1})$
  - ▶ Produzir  $c_0, c_1, \dots, c_t$
- ▶ Decifração exige que  $F$  seja invertível
- ▶ Expansão de texto cifrado é somente de 1 bloco

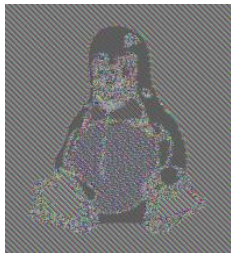
## Modo CBC – Cipher block chaining



# Modo ECB – Eletronic Codebook

- ▶  $Enc_k(m_1, \dots, m_t) = F_k(m_1), \dots, F_k(m_t)$
- ▶ Determinístico
  - ▶ Não é seguro contra CPA
- ▶ Possível diferenciar o que é igual
  - ▶ Se  $m_i = m_j$ ,  $F_k(m_i) = F_k(m_j)$
  - ▶ Não tem nem mesmo encriptações indiscerníveis

## Efeito do modo ECB



- ▶ Fonte: [https://en.wikipedia.org/wiki/Block\\_cipher\\_mode\\_of\\_operation](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation)