# Bacharelado em Ciência da Computação

CCMP3079 Segurança de Redes de Computadores

## Atividade Cap. 02

## Prof. Sérgio Mendonça

Nome Completo: Luan Valentino Sampaio Marques

**Data:** 17/10/2023

Universidade Federal do Agreste de Pernambuco
Av. Bom Pastor s/n - Boa Vista
55292-270 Garanhuns/PE
T +55 (87) 3764-5500
http://www.ufape.edu.br

# Questões

1. Responda os questionamentos a seguir:

   (a) **Por que é importante estudar a cifra de Feistel?**

   A cifra de Feistel é importante porque fornece uma estrutura robusta e versátil para a construção de cifras de bloco seguras. Ela é a base para muitos algoritmos de criptografia modernos, como o DES (Data Encryption Standard). O design de Feistel permite a utilização de funções de permutação e substituição que ajudam a alcançar um bom equilíbrio entre segurança e eficiência. Além disso, a cifra de Feistel é conhecida por sua capacidade de criar sistemas criptográficos que podem ser facilmente adaptados e modificados para diferentes necessidades de segurança.

   (b) **Qual é a diferença entre uma cifra de bloco e uma cifra de fluxo?**

   A principal diferença entre cifras de bloco e cifras de fluxo é a forma como elas processam os dados. Uma cifra de bloco encripta e decripta os dados em blocos de tamanho fixo, geralmente em tamanhos como 64 ou 128 bits. Ela transforma um bloco de texto claro em um bloco de texto cifrado de tamanho igual. Por outro lado, uma cifra de fluxo opera em fluxos contínuos de dados, encriptando ou decriptando um bit ou byte de cada vez. Enquanto cifras de bloco são mais adequadas para sistemas com dados em grandes blocos, cifras de fluxo são preferíveis para aplicações em tempo real, onde os dados são processados continuamente.

   (c) **Por que não é prático usar uma cifra de substituição reversível qualquer do tipo mostrado na Tabela 3.1?**

   Cifras de substituição simples, como as mostradas na Tabela 3.1, não são práticas porque são vulneráveis a ataques de criptanálise, como análise de frequência. Em cifras de substituição, cada letra do alfabeto é mapeada para outra letra de forma fixa, o que facilita a identificação de padrões e frequências no texto cifrado. Isso torna mais fácil para um atacante realizar uma análise estatística e decifrar a mensagem sem conhecer a chave. Além disso, cifras de substituição não oferecem uma difusão significativa, o que limita sua segurança.

   (d) **O que é uma cifra de produto?**

   Uma cifra de produto é um tipo de cifra que combina duas ou mais operações criptográficas, como substituição e permutação, para criar um sistema de encriptação mais seguro. Em vez de usar uma única operação de criptografia, a cifra de produto aplica várias operações em sequência, o que aumenta a complexidade e a segurança do sistema. A cifra de Feistel é um exemplo de cifra de produto, pois combina operações de substituição e permutação de maneira iterativa para fortalecer a segurança.

   (e) **Qual é a diferença entre difusão e confusão?**

Difusão e confusão são dois conceitos fundamentais em criptografia que ajudam a garantir a segurança de um sistema criptográfico. Difusão refere-se ao processo de espalhar o impacto das alterações no texto claro por todo o texto cifrado, de modo que cada parte do texto cifrado dependa de muitas partes do texto claro. Confusão refere-se à complexidade do relacionamento entre o texto cifrado e a chave criptográfica, tornando difícil para um atacante determinar a chave com base no texto cifrado. Em resumo, difusão assegura que pequenas alterações no texto claro resultem em grandes mudanças no texto cifrado, enquanto confusão assegura que o texto cifrado não revele informações diretas sobre a chave.

(f) **Que parâmetros e escolhas de projeto determinam o algoritmo real de uma cifra de Feistel?**

O algoritmo real de uma cifra de Feistel é determinado por vários parâmetros e escolhas de projeto, incluindo:

- **Número de Rodadas:** O número de rodadas de permutação e substituição aplicadas à cifra. Um maior número de rodadas geralmente aumenta a segurança.
- **Função de Feistel:** A função utilizada para a operação de permutação e substituição dentro da estrutura Feistel. A segurança da cifra depende fortemente da eficácia dessa função.
- **Tamanho da Chave:** O tamanho da chave usada para a encriptação e decriptação, que deve ser suficientemente grande para garantir a segurança.
- **Estrutura dos Sub-blocos:** O tamanho dos sub-blocos em que o texto é dividido durante o processo de criptografia.

(g) **Explique o efeito avalanche.**

O efeito avalanche é uma propriedade desejável em algoritmos criptográficos, onde uma pequena alteração no texto claro ou na chave resulta em uma grande mudança no texto cifrado. Em outras palavras, se um bit do texto claro ou da chave for alterado, a mudança resultante no texto cifrado deve ser significativa e não previsível. Isso ajuda a garantir que padrões no texto claro não sejam facilmente detectáveis no texto cifrado, aumentando a segurança geral do sistema criptográfico. O efeito avalanche é crucial para a resistência contra ataques de criptanálise e para a proteção dos dados encriptados.

2. Qual(is) dos recursos abaixo estão presentes no projeto da rede de Feistel? Explique.

(a) **Tamanho do bloco e da chave;**

(b) **Função da rodada;**

(c) **Gerador de sub-chaves;**

(d) **Todas as alternativas.**

**Resposta:**

A resposta correta é **(d) Todas as alternativas**. A rede de Feistel utiliza todos os recursos listados:

- **Tamanho do bloco e da chave:** A cifra de Feistel opera sobre blocos de texto de tamanho fixo, e o tamanho da chave determina a complexidade e a segurança do sistema criptográfico. O tamanho do bloco e da chave são parâmetros críticos que afetam a segurança e a eficiência da cifra.
- **Função da rodada:** A função da rodada é um componente essencial da rede de Feistel. Ela é aplicada em cada rodada do processo de criptografia e é responsável por realizar operações de permutação e substituição nos dados. A segurança da cifra depende fortemente da eficácia e complexidade dessa função.
- **Gerador de sub-chaves:** Em muitas implementações de cifras de Feistel, uma função de geração de sub-chaves é usada para produzir uma série de sub-chaves a partir da chave principal. Essas sub-chaves são usadas em diferentes rodadas do processo de criptografia, aumentando a segurança ao garantir que cada rodada use uma chave diferente.

3. Qual é o tamanho do texto claro no Data Encryption Standard (DES)? Explique.

   - (a) 57;
   - (b) 48;
   - (c) 32;
   - (d) 64.

   **Resposta:**

   A resposta correta é **(d) 64**. No Data Encryption Standard (DES), o tamanho do texto claro (ou seja, o bloco de dados que é criptografado de uma vez) é de 64 bits. O DES opera em blocos de 64 bits, o que significa que cada bloco de texto claro é dividido em partes de 64 bits antes de ser processado pela cifra.

4. A cifra de Feistel do algoritmo de encriptação utilizada no Data Encryption Standard (DES) utiliza quantos S-boxes? Explique.

   - (a) 8;
   - (b) 7;
   - (c) 6;
   - (d) 5.

   **Resposta:**

   A resposta correta é **(a) 8**. O DES utiliza 8 S-boxes (caixas de substituição) em sua estrutura de cifra de Feistel. Cada S-box é responsável por realizar uma substituição não-linear dos bits, contribuindo para a complexidade e a segurança da cifra. Cada um dos 8 S-boxes opera em uma entrada de 6 bits e produz uma saída de 4 bits.

5. O Data Encryption Standard possui uma chave de 56 bits, o que torna possível um espaço de $2^{56}$ chaves possíveis. Essa sentença trata de ataque de... Explique.

   - (a) Tempo;
   - (b) Matemático;
   - (c) Força-Bruta;

- (d) DoS.

**Resposta:**

A resposta correta é **(c) Força-Bruta**. A descrição refere-se a um ataque de força-bruta, onde um atacante tenta todas as possíveis chaves para descobrir a chave correta. No caso do DES, com uma chave de 56 bits, existem $2^{56}$ possíveis combinações de chaves, e um ataque de força-bruta envolve testar cada uma dessas combinações até encontrar a correta.

6. **Demonstre, através de um exemplo, como realizar a cifragem de 16 bits (dois caracteres), em 2 rounds, em seguida, decifre o texto cifrado. Explique o processo passo a passo. Forneça um código Python/SageMath com sua solução.**

**Resposta:**

Vamos usar uma cifra de Feistel simplificada para cifrar e decifrar um bloco de 16 bits em 2 rodadas. Suponhamos que a função de rodada seja uma função XOR simples com a chave. O exemplo a seguir ilustra o processo.

**Exemplo:**

- Texto claro: 'AB' (em hexadecimal: 0x4142) - Chaves para as rodadas: $K_1 = 0x1234$ e $K_2 = 0x5678$

**Processo de Cifração:**

1. **Divida o texto claro em dois blocos de 8 bits:**

$$L_0 = 0x41, \quad R_0 = 0x42$$

2. **Rodada 1:**

$$L_1 = R_0 = 0x42$$

$$R_1 = L_0 \oplus F(R_0, K_1)$$

onde $F(R, K) = R \oplus K$ (a função de rodada é uma XOR com a chave)

$$F(0x42, 0x1234) = 0x42 \oplus 0x1234 = 0x1276$$

$$R_1 = 0x41 \oplus 0x1276 = 0x1237$$

3. **Rodada 2:**

$$L_2 = R_1 = 0x1237$$

$$R_2 = L_1 \oplus F(R_1, K_2)$$

$$F(0x1237, 0x5678) = 0x1237 \oplus 0x5678 = 0x445F$$

$$R_2 = 0x42 \oplus 0x445F = 0x445D$$

4. **Texto cifrado:**

$$\text{Texto cifrado} = (L_2, R_2) = (0x1237, 0x445D)$$

**Processo de Decifraçao:**

1. **Divida o texto cifrado em dois blocos de 8 bits:**

$$L_2 = 0x1237, \quad R_2 = 0x445D$$

2. **Rodada 2:**

$$L_1 = R_2 = 0x445D$$
$$R_1 = L_2 \oplus F(R_2, K_2)$$
$$F(0x445D, 0x5678) = 0x445D \oplus 0x5678 = 0x0275$$
$$R_1 = 0x1237 \oplus 0x0275 = 0x1202$$

3. **Rodada 1:**

$$L_0 = R_1 = 0x1202$$
$$R_0 = L_1 \oplus F(R_1, K_1)$$
$$F(0x1202, 0x1234) = 0x1202 \oplus 0x1234 = 0x0036$$
$$R_0 = 0x445D \oplus 0x0036 = 0x445B$$

4. **Texto claro recuperado:**

$$\text{Texto claro} = (L_0, R_0) = (0x1202, 0x445B)$$

**Código Python:**

```
def feistel_round(L, R, K):
    return (R, L ^ (R ^ K))

def encrypt_decrypt(text, keys):
    L = text >> 8
    R = text & 0xFF

    # Round 1
    L, R = feistel_round(L, R, keys[0])
    # Round 2
    L, R = feistel_round(L, R, keys[1])

    return (L << 8) | R

# Exemplo
plaintext = 0x4142
keys = [0x1234, 0x5678]
ciphertext = encrypt_decrypt(plaintext, keys)
decrypted = encrypt_decrypt(ciphertext, keys[::-1])

print(f"Texto cifrado: {ciphertext:04X}")
print(f"Texto decifrado: {decrypted:04X}")
```

7. **Considere uma cifra de Feistel composta de 16 rodadas com tamanho de bloco de 128 bits e tamanho de chave de 128 bits. Suponha que, para determinado $k$, o algoritmo de escalonamento de chave defina valores as oito primeiras chaves de rodada, $k_1, k_2, \ldots, k_8$, e depois estabeleça $k_9 = k_8, k_{10} = k_7, k_{11} = k_6, \ldots, k_{16} = k_1$. Admita que você tenha um texto cifrado $S$. Explique como, com acesso a um oráculo de encriptação, você pode**

**decifrar $c$ e determinar $m$ usando apenas uma única consulta a ele. Isso mostra que tal cifra é vulnerável a um ataque de texto claro escolhido. (Um oráculo de encriptação pode ser imaginado como um dispositivo que, dado um texto claro, retorna o texto cifrado correspondente. Os detalhes internos do dispositivo não são conhecidos, e você não pode abri-lo. Você só consegue obter informações do oráculo fazendo consultas a ele e observando suas respostas.)**

**Resposta:**

Para explorar a vulnerabilidade da cifra de Feistel descrita, você pode usar um oráculo de encriptação da seguinte maneira:

1. **Escolha um texto claro $M$ e consulte o oráculo para obter o texto cifrado $S$.**

$$S = \text{Oráculo}(M)$$

2. **Utilize o conhecimento de que as chaves de rodada são repetidas, ou seja, a sequência de chaves é simétrica: $k_9 = k_8$, $k_{10} = k_7$, etc. Isso implica que a cifra de Feistel pode ser simplificada ao realizar a cifragem e a decifragem com as mesmas chaves.**

3. **Escolha um texto claro $M_1$ e consulte o oráculo para obter $S_1$. Então escolha um texto claro $M_2$ e consulte o oráculo para obter $S_2$.**

4. **Comparando $S_1$ e $S_2$ com $S$, você pode deduzir a chave de rodada por meio da análise das diferenças nos textos cifrados.**

5. **Com o acesso ao oráculo e a capacidade de consultar diferentes textos claros, você pode explorar o padrão repetitivo das chaves e deduzir a chave completa. Isso demonstra que a cifra é vulnerável a um ataque de texto claro escolhido, pois a estrutura simétrica das chaves permite a recuperação simplificada da chave com uma única consulta.**