Cifra de Bacon

Nessa seção veremos um primeiro exemplo de Esteganografia: a **Cifra de Bacon**.

> “ Ela tem esse nome pois foi criada por Francis Bacon em 1605.

A ideia por trás dessa cifra, diferente de uma cifra criptográfica comum, é esconder a mensagem secreta por meio de um texto legível, apenas mudando a grafia de suas letras.

Essa é uma cifra de substituição, e cada letra é representada por um conjunto de 5 caracteres binários ('a' e 'b' ou '0' e '1'). O **alfabeto de substituição** da Cifra de Bacon possui dois modelos:

- **A cifra de 24 letras:** É a original. Nela, os pares de caracteres (I,J) e (U,V) não possuem distinção.

```
A = aaaaa  I/J = abaaa    R = baaaa
B = aaaab    K = abaab    S = baaab
C = aaaba    L = ababa    T = baaba
D = aaabb    M = ababb  U/V = baabb
E = aabaa    N = abbaa    W = babaa
F = aabab    O = abbab    X = babab
G = aabba    P = abbba    Y = babba
H = aabbb    Q = abbbb    Z = babbb
```

- **A cifra de 26 letras:** A segunda versão da cifra. Agora todas as letras possuem um código único.

```
A = aaaaa  I = abaaa  Q = baaaa  Y = bbaaa
B = aaaab  J = abaab  R = baaab  Z = bbaab
C = aaaba  K = ababa  S = baaba
D = aaabb  L = ababb  T = baabb
E = aabaa  M = abbaa  U = babaa
F = aabab  N = abbab  V = babab
G = aabba  O = abbba  W = babba
H = aabbb  P = abbbb  X = babbb
```

Dessa forma, substituímos o 'A' da mensagem secreta por 'aaaaa', um 'B' por 'aaaab' e assim por diante. Por exemplo, digamos que queremos esconder a mensagem `fly you fools`. Assim, a

substituição das letras será da forma abaixo, usando a cifra de 26 letras:

```
texto original: F     L     Y     Y     O     U     F     O     O     L     S
texto cifrado:  aabab ababb bbaaa bbaaa abbba babaa aabab abbba abbba ababb baaba
```

Com isso, podemos usar esse padrão e esconder em uma mensagem comum de tamanho maior ou igual ao texto cifrado, como `according to all known laws of aviation, there is no way a bee should be able to fly` .

Assim, removendo os espaços e pontuações para facilitar a associação, podemos esconder a mensagem no texto de várias formas, como associar letras **maiúsculas** ao 'a' e **minúsculas** ao 'b', associar a letras com ou sem **itálico** ou até com duas **fontes diferentes**. Para esse exemplo, usaremos maíusculas e minúsculas.

```
falsa mensagem: accordingtoallknownlawsofaviationthereisnowayabeeshouldbeabletofly
texto cifrado:  aababababbbbaaabbaaaabbbababaaaababababbbaabbbaababbbaaba
mensagem final: acCoRdInGTOAllkNOwnlaWSOfAvIatioNtHeREIsnOWAyaBeESHouLdbeabletofly
```

Voltando à formatação original da mensagem, temos: `acCoRdInG TO All kNOwn laWS Of AvIatioN, tHeRE Is nO WAy a BeE SHouLd be able to fly` . Com isso, quem interceptar essa mensagem esteganográfica, não vai imaginar que ela possui algo além de uma escrita engraçada.

Identificando

Em uma primeira observação, uma mensagem oculta pela cifra de Bacon pode ser identificada pela mudança alternada dos padrões das letras: maíuscula e minúscula, itálico, fonte, ou até a alternância explícita de duas letras.

Outra abordagem, seria uma análise de frequência. Como a cifra de Bacon é um tipo de cifra de substituição, ela é sensível a uma análise de frequência.

Solucionando

Para solucionar desafios que contém Cifra de Bacon, o primeiro passo é identificar o modo como a mensagem foi escondida. Depois disso, é preciso percorrer o texto e transformá-lo em sequências de 'a's e 'b's.

Assim, você pode guardar o alfabeto de substituição com um dicionário e usá-lo para substituir os blocos de 5 caracteres pela letra correspondente.

Por exemplo, podemos implementar isso com um código em Python:

```
bacon_to_letter_26 = {
    'aaaaa':'A', 'aaaab':'B', 'aaaba':'C', 'aaabb':'D', 'aabaa':'E',
    'aabab':'F', 'aabba':'G', 'aabbb':'H', 'abaaa':'I', 'abaab':'J',
    'ababa':'K', 'ababb':'L', 'abbaa':'M', 'abbab':'N', 'abbba':'O',
    'abbbb':'P', 'baaaa':'Q', 'baaab':'R', 'baaba':'S', 'baabb':'T',
    'babaa':'U', 'babab':'V', 'babba':'W', 'babbb':'X', 'bbaaa':'Y',
    'bbaab':'Z'
}

def format(text, a='a', b='b'):
    """Format a steganographic text to a binary sequence"""
    formated_text = ''
    for c in text:
        if not c.isalpha():
            continue
        if c.istitle():
            formated_text += b
        else:
            formated_text += a
    return formated_text

def decode(text, a='a', b='b', bacon_alpha=bacon_to_letter_26):
    """Decode a encrypted Bacon cipher text"""
    cipher = format(text, a=a, b=b)
    output = ''
    while len(cipher) >= 5:
        token, cipher = cipher[:5], cipher[5:]
        if token in bacon_alpha:
            output += bacon_alpha[token]
        else:
            break
    return output

if __name__ == '__main__':
    input  = input()
    output = decode(input)
    print(output)
```

Exercícios

WeChall: Baconian

WeChall: Bacon Returns

Referências

- GeeksforGeeks
- Practical Cryptography

Revision #3
Created Sat, Oct 6, 2018 7:03 PM by Andrew
Updated Sat, Oct 6, 2018 7:05 PM by Andrew