Шифрование Гаммированием

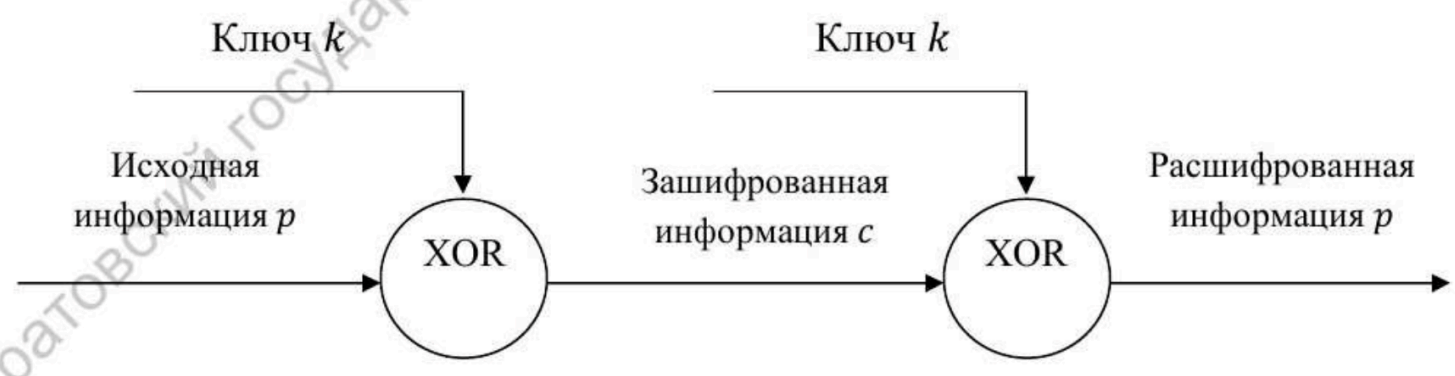
Gagik T. Papikyan

RUDN University, Moscow, Russian Federation

Шифрование Гаммированием

Простой способ гаммирования

- Обратимость
- Ключ равный сообщению по длине



Шифрование конечной гаммой

- Отсутствие обратимости
- Короткий ключ

Например, зашифруем слово «ПРИКАЗ» («16 17 09 11 01 08») гаммой «ГАММА» («04 01 13 13 01»). Будем использовать операцию побитового сложения по модулю 33 ($mod\ 33$). Получаем:

$c_1 = 16 + 4 (mod\ 33) = 20$ $\quad$ $c_4 = 11 + 13 (mod\ 33) = 24$

$c_2 = 17 + 1 (mod\ 33) = 18$ $\quad$ $c_5 = 1 + 1 (mod\ 33) = 2$

$c_3 = 9 + 13 (mod\ 33) = 22$ $\quad$ $c_6 = 8 + 4 (mod\ 33) = 12.$

Криптограмма: «УСХЧБЛ» («20 18 22 24 02 12»).

Связь с шифром Цезаря

- Шифрование конечной гаммой использует идею шифра Цезаря

Например, зашифруем слово «ПРИКАЗ» («16 17 09 11 01 08») гаммой «ГАММА» («04 01 13 13 01»). Будем использовать операцию побитового сложения по модулю 33 ($mod\ 33$). Получаем:

$$c_1 = 16 + 4 (mod\ 33) = 20 \qquad c_4 = 11 + 13 (mod\ 33) = 24$$

$$c_2 = 17 + 1 (mod\ 33) = 18 \qquad c_5 = 1 + 1 (mod\ 33) = 2$$

$$c_3 = 9 + 13 (mod\ 33) = 22 \qquad c_6 = 8 + 4 (mod\ 33) = 12.$$

Криптограмма: «УСХЧБЛ» («20 18 22 24 02 12»).

Реализация шифрования конечной гаммой

```
const alphabet = 'qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM !'
const reversedAlphabet = alphabet.split('').reverse().join('')

function gammaCipher(msg, gamma, decode = false){
    const fullGamma = gamma.repeat(Math.ceil(msg.length / gamma.length))

    const msgCodes = msg.split('').map( char=> ( decode ?  reversedAlphabet : alphabet ).indexOf(char) )
    const fullGammaCodes = fullGamma.split('').map( char=>alphabet.indexOf(char) )

    const resCodes = msgCodes.map( (msgCode, idx) => (msgCode+fullGammaCodes[idx])%alphabet.length )
    const res = resCodes.map( code=>( decode ?  reversedAlphabet : alphabet )[code] ).join('')

    return res
}
```