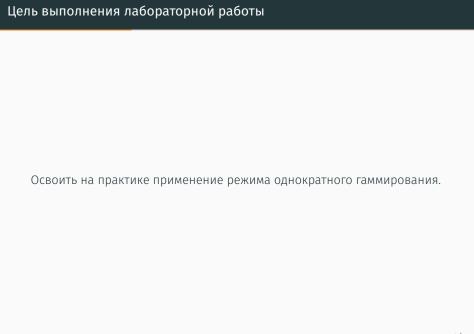
Защита лабораторной работы №7. Элементы криптографии. Однократное гаммирование

Асеинова Елизавета 2022 Oct 5th

RUDN University, Moscow, Russian Federation

Результат выполнения

лабораторной работы №7



```
[24] def shest(message):
    return ' '.join(hex(ord(i))[2::] for i in message)

def rand_key(s):
    return ' '.join(random.choice(string.ascii_letters + string.digits) for _ in range(s))

def code(message,key):
    return ' '.join(chr(a^b) for a,b in zip(message, key))

def encode(message, enc):
    return ' '.join(chr(a^b) for a,b in zip(message, enc))
```

Figure 1: Функции

```
[25] key = rand_key(len(message))
hex_key = shest(key)
print('Рандомный ключ: ', key)
print('Ключ в шестнадцатеричном представлении: ', hex_key)

Рандомный ключ: 1 m Q K 5 0 c e K Q P j d C o M w 4 e 1 a r
Ключ в шестнадцатеричном представлении: 6c 20 6d 20 51 20 4b 20 35 20 4f 20 63 20 65 20 4b 20 51 20 50 2
```

Figure 2: Задание ключа

```
С mess = code([ord(i) for i in message],[ord(i) for i in key])
hex_c_mess = shest(c_mess)
print('Зашифрованный текст в 16ном представлении: ', hex_c_mess)
Зашифрованный текст в 16ном представлении: 44d 20 0 20 470 20 41e 20 463 20 46b 20 477 20 0 20 426 20 41e 20 47b 20 41e 20 45f 20 41e 20 47b 20 41e 20 47b 20 41e 20 45f 20 41e 20 47b 20
```

Figure 3: Зашифрованный текст

```
[27] key2 = encode([ord(i) for i in message],[ord(i) for i in c_mess])
mess = code([ord(i) for i in message],[ord(i) for i in key2])
print(mess)

→ ♦ Н 0 / х " ♦ ё 0 _ 0 0 Даѣ) КАВ
```

Figure 4: Расшифрованный текст



В ходе работы мы освоили на практике применение режима однократного гаммирования.