Защита лабораторной номер 2

по предмету мат. основы защиты информации

Дидусь К.В.

Российский университет дружбы народов, Москва, Россия

Вводная часть

Актуальность

- Изучение базовых принципов шифрования
- Важность знания основ шифрования для работы в сфере информационных технологий

Объект и предмет исследования

- Шифры перестановки
  - Шифр вертикальной перестановки
  - Шифрование с помощью решеток
  - Шифр Виженера

Цели и задачи

- Ознакомиться с шифрами перестановки
- Реализовать шифры на любом языке программирования

Материалы и методы

- Национальный Открытый Университет “ИНТУИТ”
- Язык программирования Python

Выполнение лабораторной работы

Шифр вертикальной перестановки

```python
# Vigenere

def generateKey(string, key):
    key = list(key)
    if len(string) == len(key):
        return(key)
    else:
        for i in range(len(string) -len(key)):
            key.append(key[i % len(key)])
    return("" . join(key))

def encryption(string, key):
    encrypt_text = []
    for i in range(len(string)):
        x = (ord(string[i]) +ord(key[i])) % 26
        x += ord('A')
        encrypt_text.append(chr(x))
    return("" . join(encrypt_text))

if __name__ == "__main__":
    string = input("Enter the message: ")
    keyword = input("Enter the keyword: ")
    key = generateKey(string, keyword)
    encrypt_text = encryption(string,key)
    print("Encrypted message:", encrypt_text)

```

Шифрование решетками

```
msg = list()
for char in msg_str:
    if char == ' ': continue
    msg.append(char)
msg_split = list()
for i in range(0, len(msg), n):
    msg_split.append(msg[i:i+n])

code = list([])
for i in range(n):
    code.append([])
    for j in range(m):
        code[i].append(msg_split[j][i])

d = dict()
p = list(password)
for i in range(n):
    d[p[i]] = code[i]

p.sort()

sorted_code = list()
for char in p:
    sorted_code.append(d[char])

final_code = ""
for i in range(n):
    for j in range(m):
        final_code = final_code + sorted_code[i][j]

print("Encoded message: ",final_code)
```

Шифр Виженера

```
64  def encrypt(k,msg):
65      print("encrypting...")
66      m = np.arange(1,(k**2)+1)
67      m = m.reshape(k,k)
68      m_upper = np.hstack((m,rotate_clockwise(m,1)))
69      m_bottom = np.hstack((rotate_clockwise(m,3),rotate_clockwise(m,2)))
70      m_final = np.vstack((m_upper,m_bottom))
71      index_arr = find_net(k,m_final)
72
73      msg = list(msg)
74
75      arr = [[]]
76      for g in range(k**2-1):
77          arr.append([])
78
79      count = 0
80      for char in msg:
81          if char == ' ':
82              continue
83          if count == k**2:
84              index_arr = rotate_index(index_arr,k)
85              count = 0
86          i,j = index_arr[count]
87          count += 1
88          arr[i].insert(j,char)
89      final_code = ""
90      for i in range(len(arr)):
91          for j in range(len(arr)):
92              final_code = final_code + arr[i][j]
93      print("Encoded message: ",final_code)
94
95
```

Результаты

- Ознакомился с шифрами простой замены: шифр Цезаря, шифр Атбаш

Результаты

- Ознакомился с шифрами простой замены: шифр Цезаря, шифр Атбаш
- Программно реализовал шифры с помощью языка программирования Python