Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Акопян Изабелла Арменовна

28 октября, 2023, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Выполнение лабораторной работы

Гаммирование

Гаммирование – это наложение (снятие) на открытые (зашифрованные) данные криптографической гаммы, т.е. последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

Алгоритм взлома

Шифротексты обеих телеграмм можно получить по формулам режима однократного гаммирования:

$$C_1 = P_1 \oplus K$$
$$C_2 = P_2 \oplus K$$

Алгоритм взлома

Открытый текст можно найти, зная шифротекст двух телеграмм, зашифрованных одним ключом. Для это оба равенства складываются по модулю 2. Тогда с учётом свойства операции XOR получаем:

$$C_1 \oplus C_2 = P_1 \oplus K \oplus P_2 \oplus K = P_1 \oplus P_2$$

Алгоритм взлома

Предположим, что одна из телеграмм является шаблоном — т.е. имеет текст фиксированный формат, в который вписываются значения полей. Допустим, что злоумышленнику этот формат известен. Тогда он получает достаточно много пар $C_1 \oplus C_2$ (известен вид обеих шифровок). Тогда зная P_1 имеем:

$$C_1 \oplus C_2 \oplus P_1 = P_1 \oplus P_2 \oplus P_1 = P_2$$

Схема работы алгоритма

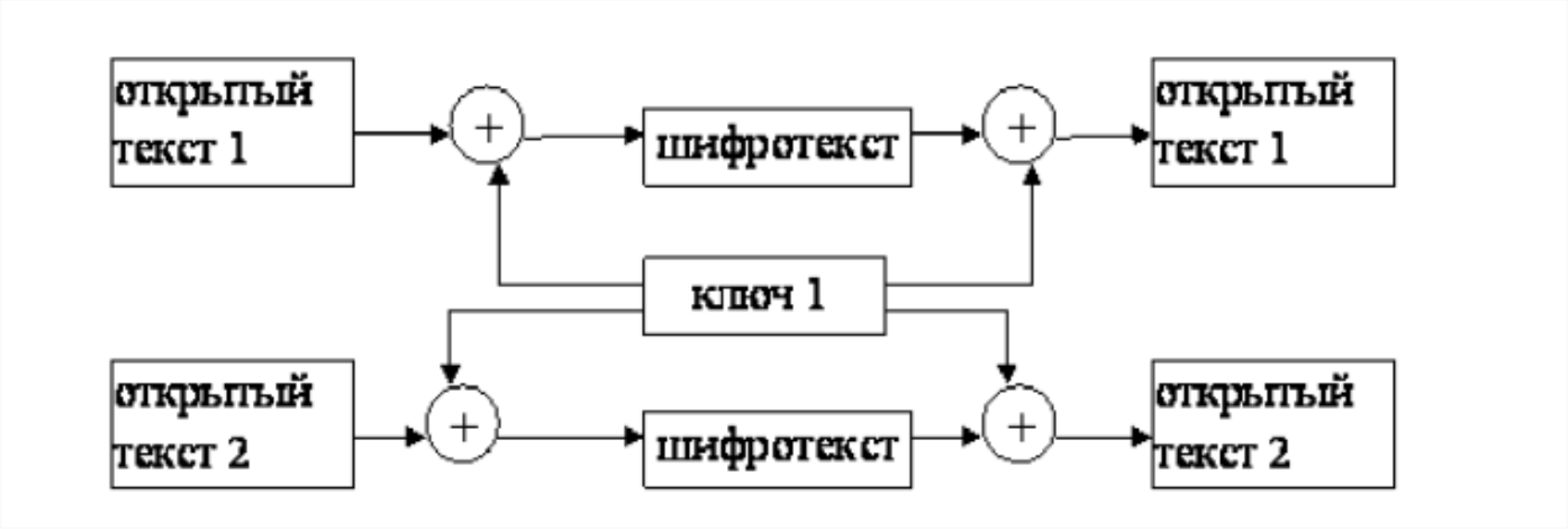

Figure 1: Работа алгоритма гаммирования

Пример работы программы

```
def shifr(P1, gamma):
  dicts = {
    "а": 1, "б": 2, "в": 3, "г": 4, "д": 5, "е": 6, "ё": 7, "ж": 8, "з": 9, "и": 10,
    "й": 11, "к": 12, "л": 13, "м": 14, "н": 15, "о": 16, "п": 17, "р": 18, "с": 19, "т": 20,
    "у": 21, "ф": 22, "х": 23, "ц": 24, "ч": 25, "ш": 26, "щ": 27, "ъ": 28, "ы": 29, "ь": 30, "э": 31, "ю": 32, "я": 33,
    "А": 34, "Б": 35, "В": 36, "Г": 37, "Д": 38, "Е": 39, "Ё": 40, "Ж": 41, "З": 42, "И": 43,
    "Й": 44, "К": 45, "Л": 46, "М": 47, "Н": 48, "О": 49, "П": 50, "Р": 51, "С": 52, "Т": 53,
    "У": 54, "Ф": 55, "Х": 56, "Ц": 57, "Ч": 58, "Ш": 59, "Щ": 60, "Ъ": 61, "Ы": 62, "Ь": 63, "Э": 64, "Ю": 65, "Я": 66,
    "1": 67, "2": 68, "3": 69, "4": 70, "5": 71, "6": 72, "7": 73, "8": 74, "9": 75, "0": 76
}
  dicts2 = {v: k for k,v in dicts.items()}
  text = P1
  digits_text = []
  digits_gamma = []

  for i in text:
    digits_text.append(dicts[i])
  print("Числа текста: ",digits_text)

  for i in gamma:
    digits_gamma.append(dicts[i])
  print("Числа гаммы: ", digits_gamma)

  digits_result = []
  ch = 0
  for i in text:
    try:
      a = dicts[i] + digits_gamma[ch]
    except:
```

Figure 2: Работа алгоритма взлома ключа

```
[74] shifr(P1, gamma)

     Числа текста:  [34, 12, 16, 17, 33, 15, 43, 9, 1, 2, 6, 13, 13, 1]
     Числа гаммы:  [3, 1, 16, 17, 17, 1, 18]
     Числа шифротекста:  [37, 13, 32, 34, 50, 16, 61, 12, 2, 18, 23, 30, 14, 19]
     Шифротекст:  ГлюАПоЬкбрхьмс
     Расшифрованный текст:  АкопянИзабелла
```

Выводы

Результаты выполнения лабораторной работы

Я овладела навыками использования режима однократного гаммирования на практике, применяя его для кодирования разнообразных исходных текстов с использованием одного и того же ключа.

Мною было разработано приложение, которое способно зашифровывать и расшифровывать тексты в режиме однократного гаммирования, а также определять тип шифротекста при наличии известного ключа.