Основы информационной безопасности

Лабораторная работа № 7. Элементы криптографии и однократное гаммирование

Подлесный Иван Сергеевич

Содержание

# Цель работы

Освоить на практике применение режима однократного гаммирования

# Задание

Нужно подобрать ключ, чтобы получить сообщение «С Новым Годом, друзья!». Требуется разработать приложение, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования. Приложение должно:

1. Определить вид шифротекста при известном ключе и известном открытом тексте.
2. Определить ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста.

# Выполнение лабораторной работы

Создадим функцию generate\_key которая будет генерировать случайный ключа(составляется выбором из букв Латиницы больших и спецсимволов ), cypher – принимает на вход текст и ключ, а затем осуществляет посимвольное сложение по модулю 2, get\_partial\_key – подбирает точную часть ключа для известного фрагмента сообщения, а затем оставшуюся часть выбирает случайным образом:

#include <iostream>  
#include <cstring>  
#include <string>  
#include <windows.h>  
#include <random>  
using namespace std;  
  
random\_device rd;  
mt19937 gen(rd());  
uniform\_int\_distribution<> distrib(64, 128);  
  
  
string generate\_key(string message){  
 string key = "";  
 string alphabet = "";  
 for(int i = 0; i < message.length(); i++){  
 key += char(distrib(gen));  
 }  
 return key;  
}  
  
string cypher(string message, string key){  
 string ciphered = "";  
 for(int i=0; i < message.length(); i++){  
 ciphered += message[i] ^ key[i];  
 }  
 return ciphered;  
}  
  
  
string get\_partial\_key(string part, string ciphered){  
 string p1\_key = cypher( part, ciphered) + generate\_key(ciphered.substr(7, ciphered.length()));  
 return p1\_key;  
  
}  
  
int main(){  
 setlocale(LC\_ALL, "Russian");  
 SetConsoleCP(1251);  
 SetConsoleOutputCP(1251);  
 string message = "«С Новым Годом, друзья!";  
 cout << message<<endl;  
 string key = generate\_key(message);  
 cout << "Key is "<< key << endl;  
 string ciphered = cypher(message, key);  
 cout<<"Ciphered "<< ciphered <<endl;  
 string part = message.substr(0, 15);  
 string partial\_key = get\_partial\_key(part, ciphered);  
 cout<<"Partial key "<< partial\_key<< endl;  
 cout<<"Deciphered "<< cypher(ciphered, partial\_key)<<endl;  
  
 return 0;  
}

В результате получим следующий вариант шифрования и один из вариантов прочтения текста(рис. fig. 1)

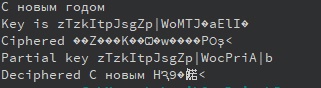


Рис. 1: Результаты работы программы

# Контрольные вопросы

1. Поясните смысл однократного гаммирования.

Гаммиирование, или Шифр XOR, — метод симметричного шифрования, заключающийся в «наложении» последовательности, состоящей из случайных чисел, на открытый текст. Если в методе шифрования используется однократная вероятностная гамма (однократное гаммирование) той же длины, что и подлежащий сокрытию текст, то текст нельзя раскрыть.

1. Перечислите недостатки однократного гаммирования.

* Если один и тот же ключ используется для шифрования нескольких сообщений, это может привести к уязвимостям. Например, если злоумышленник узнает открытый текст и соответствующий шифротекст, он может использовать эту информацию для взлома ключа.
* Однократное гаммирование не обеспечивает аутентификацию или целостность данных. Это означает, что злоумышленник может изменить шифротекст без заметных изменений в открытом тексте.

1. Перечислите преимущества однократного гаммирования.

* Однократное гаммирование обеспечивает высокий уровень конфиденциальности, поскольку шифротекст не может быть легко взломан без знания ключа.
* Однократное гаммирование обеспечивает равномерное распределение вероятностей для каждого символа в шифротексте, что делает его статистически неразличимым от случайной последовательности.
* Однократное гаммирование является простым и быстрым методом шифрования.

1. Почему длина открытого текста должна совпадать с длиной ключа?

Если в методе шифрования используется однократная вероятностная гамма (однократное гаммирование) той же длины, что и подлежащий сокрытию текст, то текст нельзя раскрыть.

1. Какая операция используется в режиме однократного гаммирования, назовите её особенности?

В режиме однократного гаммирования используется операция XOR (исключающее ИЛИ). Особенностью операции XOR является то, что она возвращает 1 только в том случае, если один из входных битов равен 1, но не оба.

1. Как по открытому тексту и ключу получить шифротекст?

Нужно побитово сложить по модулю численное представление символов в ключе и в открытом тексте.

1. Как по открытому тексту и шифротексту получить ключ?

Нужно побитово сложить по модулю численное представление символов в шифротексте и в открытом тексте.

1. В чем заключаются необходимые и достаточные условия абсолютной стойкости шифра?

Необходимые и достаточные условия абсолютной стойкости шифра:

* Ключ является фрагментом истинно случайной двоичной последовательности с равномерным законом распределения.
* Длины ключа и открытого текста совпадают.
* Ключ используется лишь один раз, после чего сразу подлежит уничтожению.

# Выводы

В результате выполнения работы были освоены практические навыки применения режима однократного гаммирования.