Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный технологический университет»

Факультет информационных технологий

Кафедра информационных систем и технологий

**Отчет к лабораторной работе**:

Исследование потоковых шифров

Выполнил:

студент 3 курса 7 группы

специальности ПОИБМС

Дубровский М.С.

Минск 2021

1. **Теоретические сведения**

Потоковый шифр (иногда говорят «поточный») – симметричный шифр, преобразующий каждый символ mi открытого текста в символ шифрованного, ci, зависящий от ключа и расположения символа в тексте. Термин «потоковый шифр» обычно используется в том случае, когда шифруемые символы открытого текста представляются одной буквой, битом или реже – байтом. Все потоковые шифры делятся на 2 класса: синхронные и асинхронные (или самосинхронизирующиеся).

Синхронные потоковые шифры (СПШ) — шифры, в которых поток ключей генерируется независимо от открытого текста и шифротекста.

Самосинхронизирующиеся потоковые шифры (асинхронные потоковые шифры (АПШ)) — шифры, в которых ключевой поток создаётся функцией ключа и фиксированного числа знаков шифротекста.

Основной задачей потоковых шифров является выработка некоторой последовательности (гаммы) для зашифрования, т.е. выходная гамма является ключевым потоком (ключом) для сообщения.

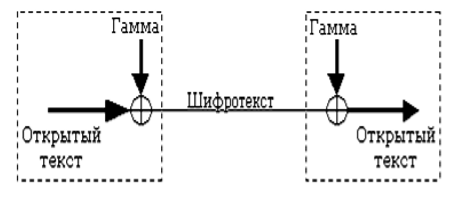


Рисунок 1.1 – Схема потокового шифра

Случайные числа (последовательности) – последовательность элементов, каждый из которых не может быть предсказан (вычислен) только на основе знания предшествующих ему элементов данной последовательности.

Псевдослучайные числа – последовательность элементов, полученная в результате выполнения некоторого алгоритма и используемая в конкретном случае вместо последовательности случайных чисел.

Алгоритм **линейного конгруэнтного генератора**, описываетя следующим рекуррентным соотношением:

xt+1 = (a\*xt + c) mod n.

а, с и n – константы. Период такого генератора (период ПСП) не превышает n.

**Генератор ПСП на основе регистров сдвига.** РС с линейной обратной связью (РСЛОС) состоит из двух частей: собственно РС и функции обратной связи. На рис. 1.2 представлена общая схема РС с линейной обратной связью. Функция обратной связи реализуется с помощью сумматоров сложения по модулю два (элементы XOR; на рис. 1.2 обозначены в виде кружочков со знаком сложения).

РСЛОС строятся на основе примитивных порождающих полиномов (многочленов). Если многочлен является неприводимым, то период ПСП при ненулевом начальном условии (ненулевом состоянии) регистра будет максимально возможным: 2L – 1.

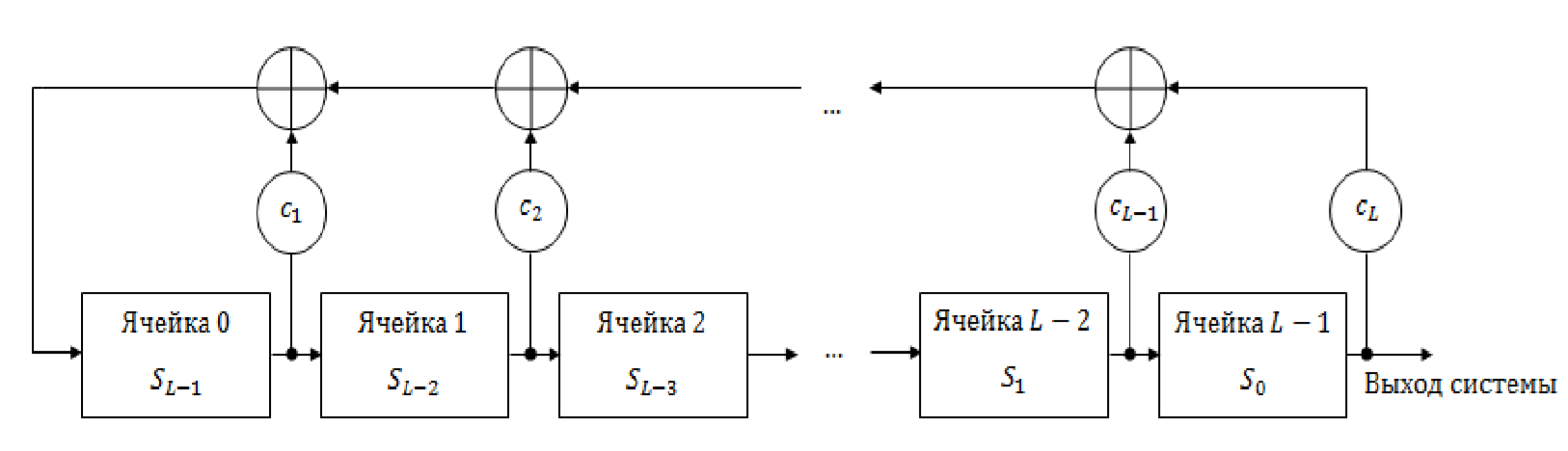


Рисунок 1.2 – Общая схема регистра сдвига с линейной обратной связью

Генератор псевдослучайных чисел на основе **алгоритма RSA** разработан для систем асимметричного зашифрования/расшифрования. Генератор ПСП на основе RSA устроен следующим образом. Последовательность генерируется с использованием соотношения:

xt = (xt-1)е mod n.

Начальными параметрами служат n, большие простые числа p и q (причем n = p\*q), целое число е, взаимно простое с произведением (р – 1)\*( q –1), а также некоторое случайное начальное значение, x0. Выходом генератора является на t-м шаге является младший бит числа xt. Безопасность генератора опирается на сложности взлома алгоритма RSA, т. е. на разложении числа n на простые сомножители.

**Алгоритм BBS** или генератором на основе квадратичных вычетов. Начальное значение x0 генератора вычисляется на основе соотношения:

x0= x2 mod n

где n, как и в генераторе на основе RSA, является произведением простых чисел p и q, однако в нашем случае эти простые числа должны быть сравнимы с числом 3 по модулю 4, т. е. при делении p и q на 4 должен получаться одинаковый остаток: 3; число x должно быть взаимно простым с n; число n называют числом Блюма. Выходом генератора на t-м шаге является младший бит числа xt:

xt = (xt-1)2 mod n.

**Алгоритм RC4**, как и любой потоковый шифр, строится на основе генератора псевдослучайных битов (генератора ПСП). На вход генератора записывается ключ, а на выходе читаются псевдослучайные биты. Длина ключа может составлять от 40 до 2048 бит.

Ядро алгоритма состоит из функции генерации ключевого потока. Другая часть алгоритма – функция инициализации, которая использует ключ переменной длины Ki для создания начального состояния генератора ключевого потока.

В основе алгоритма – размер блока или слова, определяемый параметром n. Обычно n = 8, но можно использовать и другие значения. Внутренне состояние шифра определяется массивом слов (S-блоком) размером 2n. При n = 8 элементы блока представляют собой перестановку чисел от 0 до 255.

1. **Практическая часть**

В данной лабораторной работе необходимо разработать приложения, которые должны реализовывать следующие операции:

• Приложение 1 должно реализовывать генерацию ПСП RSA, где p, q, e – 512-разрядные числа;

• Приложение 2 должно реализовывать алгоритм RC4 с ключом: 121, 14, 89, 15;

В связи с поставленными требованиями было разработано приложение реализующее алгоритм RSA на основе встроенного класса System.Security.Cryptography в C#.

Пример шифрации:

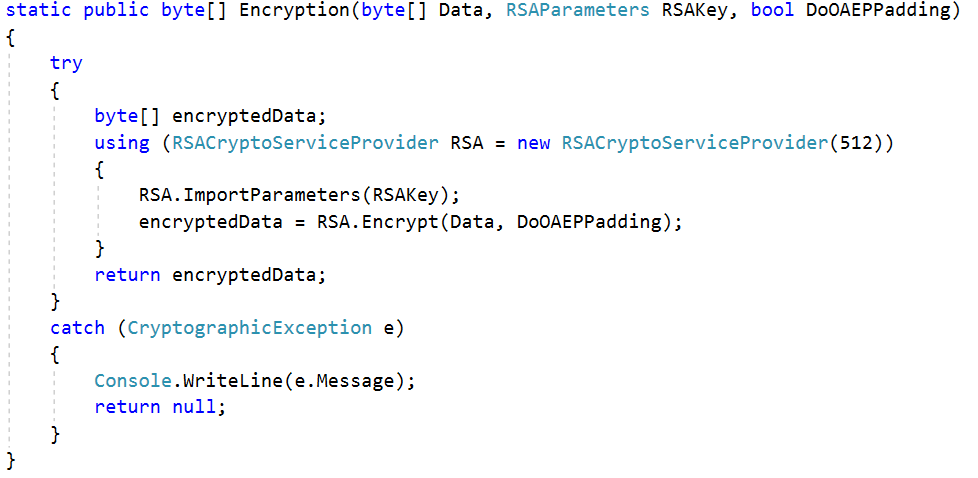


Рисунок 2.1 – Листинг шифрования

Пример дешифрации:

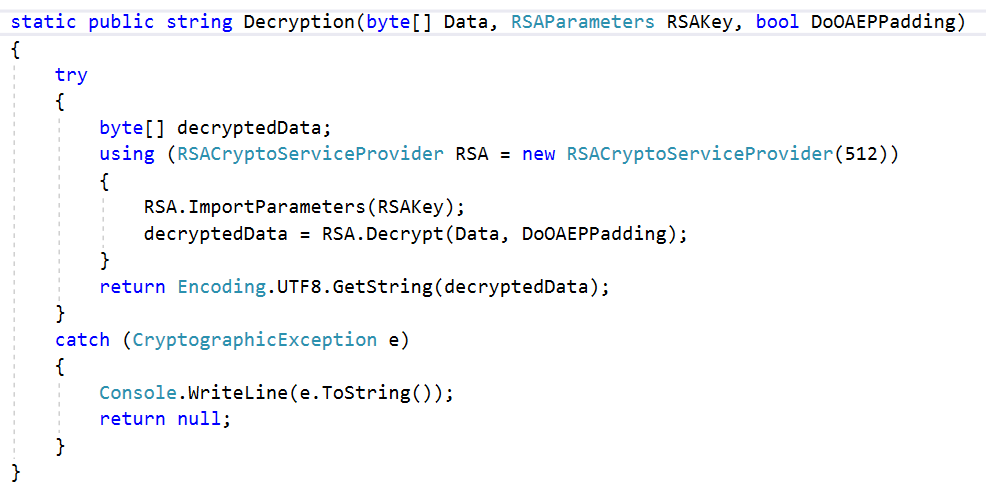


Рисунок 2.2– Листинг дешифрования

В приложении 2 было реализовано шифрование алгоритмом RC4 с помощью Python:

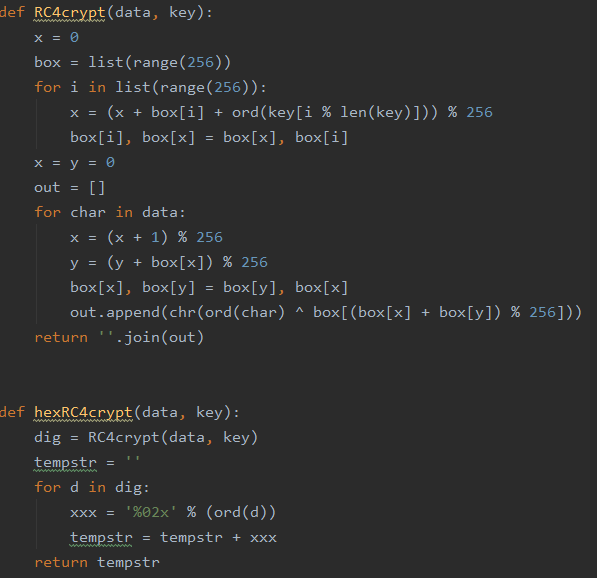


Рисунок 2.3 – Листинг шифрования

**Вывод**

В данной лабораторной работе я закрепил теоретические знания по поточным шифрам и их разновидностям. А так же разработал приложения для шифрации/дешифрации по RSA и RC4 алгоритмам.