Лабораторна робота №8

Потоковий шифр на основі генератора BBS

Мета:

Створити потоковий шифр з використанням програмного генератора псевдо-випадкових послідовностей BBS.

Обладнання:

- персональний комп'ютер з встановленою операційною системою.
- будь-яка мова програмування.

Завдання:

1. Створити потоковий шифр на основі генератора BBS.
2. Перевірити його роботу.

Література:

1. Молдовян Н.А., Молдовян А.А. Введение в криптосистемы с открытым ключом. БХВ-Петербург, 2005. – 288 с.
2. Б.Шнайер. Прикладная криптография. Протоколы, алгоритмы, исходные коды на языке С. 1996.
3. Столлингс В. Криптография и защита сетей. «Вильямс», 2001. – 672 с.

Теоретичні відомості

Програмний генератор двійкових послідовностей BBS (назву утворено від перших літер його авторів – Ленори та Мануеля Блум та Майка Шуба, Blum-Blum-Shub) вважають одним з найсильніших програмних генераторів псевдовипадкових послідовностей. Він вважається криптографічно стійким, і може використовуватися у серйозних криптографічних застосуваннях [2].

Нехай є два простих числа, p і q, причому $p \equiv q \equiv 3 \bmod 4$. Добуток цих чисел $n=pq$ називається цілим числом Блума. Оберемо ще одне випадкове число, x, взаємно просте з n та обчислимо $x_0 \equiv x \bmod n$. Це число вважається стартовим числом генератора.

Далі можна обчислити наступні біти послідовності за формулою: $x_i \equiv x_{i-1}^2 \bmod n$ та $s_i \equiv x_i \bmod 2$. Останнє визначає, що в якості виходу генератора обирається молодший біт числа x_i. Отже ми можемо записати:

$x_0 = x^2 \bmod n$
for $i = 1$ to ∞
  $x_i = (x_{i-1})^2 \bmod n$
  $s_i = x_i \bmod 2$

Найцікавішою властивістю генератора BBS є те, що для визначення значення i-го біту зовсім необов'язково знати усі попередні i-1 бітів. Для безпосереднього обчислення значення i-го біту достатньо знати p та q.

Безпека цієї схеми ґрунтується на складності розкладання n на множники. Число n можна опублікувати, так що кожен зможе генерувати біти за допомогою цього генератора. Однак поки криптоаналітик не розкладе n на множники, він не зможе передбачити вихід генератора.

Більше того, генератор BBS непередбачуваний як в правому, так і в лівому напрямках. Це означає, що отримавши послідовність бітів, криптоаналітик не зможе передбачити ні наступний, ні попередній біти послідовності. Причиною цього є не якійсь заплутаний механізм генерації, а математика розкладання n на множники.

Приклад:
p=19; q=23

$p=q\equiv 3 \bmod 4$
n=437
x=233

i	0	1	2	3	4	5	6	7
x_i	101	150	213	358	123	271	25	188
S_i	1	0	1	0	1	1	1	0

Обов'язковою умовою, що накладається на зародок x, повинно бути наступне:
а) x – просте; б) x не ділиться на p і на q.

Цей генератор повільний, але є спосіб його прискорити. Як вказано у [2], в якості бітів псевдовипадкової послідовності можна використовувати не один молодший біт, а $\log_2 m$ молодших бітів, де m – довжина числа x_i. Порівняна повільність цього генератора не дозволяє використовувати його для потокового шифрування (цей недолік зі зростанням швидкодії комп'ютерів стає менш актуальним), а от для високонадійних застосувань, як наприклад, генерування ключів, він вважається кращим за багато інших.

Однак у цій лабораторній роботі, яка лише демонструє використання потокового шифру, і де швидкодія не може бути визначальним параметром, ми будемо використовувати саме генератор BBS.

Практична частина

Складіть програму, яка б реалізовувала потоковий шифр на основі генератора BBS. Шифрування інформації повинно виконуватися за допомогою побітового XOR двійкового представлення чергового символу відкритого тексту та послідовності генератора BBS.

Оберіть два тризначних числа p, q, обчисліть модуль n та випадкове число x. n та x збережіть, оскільки їх треба передати на приймальну сторону.

За допомогою створеної програми зашифруйте та розшифруйте повідомлення.

Зробіть висновок про якість роботи Вашої системи шифрування.

Підготуйте звіт з лабораторної роботи. Звіт повинен містити: а) результати досліджень; б) протокол Ваших дій; в) код програми; г) висновок з лабораторної роботи; д) відповіді на контрольні запитання.

Контрольні запитання

1. На чому ґрунтується крипостійкість генератора BBS?
2. Які переваги і недоліки потокових шифрів?
3. Які переваги і недоліки шифрів однократного гамування?
4. Які вимоги ставляться до генераторів випадкових та псевдовипадкових послідовностей?
5. Запропонуйте кілька реалізацій генераторів випадкових послідовностей за допомогою комп'ютера.