Лабораторная работа №4

Тема «Реализация генератора псевдослучайной последовательности»

ВЫПОЛНИЛ

Олишкевич Игорь Русланович

Группа 25-ПО

Задание: изучить математические особенности псевдослучайной последовательности; выявить использование генератора; изучить использование генератора в криптографии; привести примеры.

Генератор последовательности называется случайным, если он не может быть достоверно воспроизведен, т.е. дважды запуская генератор с абсолютно одинаковыми исходными данными, будут выданы случайные различные последовательности.

Линейный конгруэнтный генератор (ЛКГ) – это последовательность чисел от 0 до 1 – m, удовлетворяющая следующему рекуррентному выражению Xk+1 = Xka + b mod m, X0 – начальное значение, a – множитель, b – приращение, m – модуль. У такого генератора период меньше m. Если a, b, m правильно выбраны, то генератор является генератором максимальной длины и имеет период m (например gcd(m, b) = 1).

Если инкремент b равен нулю, то есть генератор имеет вид: Xk+1 = Xk a mod m, получится простая последовательность, которую можно предложить для генератора с равномерным распределением. При соответствующем выборе констант a может принимать значения 75 либо 16 807 и m принимать значения 231 – 1 = 2 147 483 647, получится генератор с максимальным периодом повторения. Эти константы были предложены учеными Парком и Миллером, поэтому генератор вида: Xk+1 = Xk mod(231 – 1), называется генератором Парка-Миллера.

Основным преимуществом ЛКГ является их быстрота, за счёт малого количества операций на байт и простота реализации. Используются редко, так как являются предсказуемыми.

Нелинейные конгруэнтные генераторы квадратичного и кубического вида соответственно: Xk+1 = (aXk2 + bXk + c)mod m и Xk+1 = (aXk3 + bXk2 + cXk + d)mod m. Они обладают большей стойкостью.

Линейный регистр с обратной связью состоит из двух частей: регистра сдвига и последовательностью ответвления. В этой схеме регистр сдвига есть последовательность битов. Как только необходим следующий бит (иногда это называется clock pulse – тактовым импульсом – т.к. схема часто реализуется в аппаратном виде), все биты регистра сдвига сдвигаются направо и LFSR выдает наиболее значимый бит. При этом наименьший значимый бит определяется посредством вычисления XOR от прочих битов регистра, согласно последовательности ответвления. Теоретически, n -битный LFSR может сгенерировать псевдослучайную последовательность длиной 2n −1 бит перед зацикливанием. Для этого регистр сдвига должен побывать во всех 2n −1 внутренних состояниях.

Для того чтобы LFSR был LFSR максимальной длины, необходимо и достаточно, чтобы полином, образованный из элементов tap sequence плюс единица был примитивным полиномом по модулю 2.

Пусть p - простое и пусть 1 r , 2 r , …, t r являются различными простыми множителями для pm −1, тогда неприводимый полином f (x) ∈ Zp[x] степени m является примитивным полиномом, если и только если, для каждого i , 1 ≤ i ≤ t выполняется условие: x(pm −1)/ri ≠ 1(mod f (x)).

Полином неприводим, если он не может быть выражен как произведение двух других полиномов, за исключением произведения 1 на самого себя.

Криптографически стойкие датчики случайных чисел.

Системно-теоретический подход: криптограф создает генератор ключевого потока, у которого есть проверяемые свойства – период, распределение битов, линейная сложность и т.д. Криптограф изучает также различные методы криптоанализа и оптимизирует ДСЧ против этих атак. Этот подход выработал набор критериев для потоковых шифров. Они были сформулированы Рюппелем:

1. большой период, отсутствие повторений;
2. критерий линейной сложности: повышенная линейная сложность, локальная линейная сложность (Линейная сложность ДСЧ – это длина кратчайшего LFSR, которая может сгенерировать выход генератора; линейная сложность есть мера случайности ДСЧ);
3. статистические критерии, такие как идеальное распределение:
4. перемешивание: любой бит ключевого потока должен быть сложным;
5. преобразованием всех или большинства битов ключа;
6. рассеивание: избыточность в подструктурах должна рассеиваться; нелинейные критерии (расстояние до линейных функций, критерий лавинообразности и т.д.).

Данный список критериев подходит для всех потоковых шифров и для всех блочных шифров. Но при системно-теоретическом подходе потоковые шифры создаются таким образом, чтобы напрямую удовлетворять вышеописанным критериям.

Основная проблема подобных криптосистем, что трудно доказать какие-либо факты об их криптостойкости. Для всех предоставленных критериев не была доказана их необходимость или достаточность. Потоковый шифр может удовлетворять всем этим принципам и оказаться нестойким.

С другой стороны, взлом каждой такой системы – отдельная задача. Если бы таких шифров было много, то криптоаналитикам могут отказаться от атак.

Потоковые шифры во многом похожи на блочные шифры – для них нет доказательств стойкости. Существует набор известных способов атаки, но стойкость к ним ничего не гарантирует.

Сложно-теоретический подход: криптограф пытается использовать теорию сложности для доказательства стойкости генератора, следовательно генераторы являются более сложными; они основаны на тех же проблемах, что применяются в криптографии с открытым ключом.

Информационно-теоретический подход: предположим, у криптоаналитика есть бесконечные компьютерные ресурсы и время. Тогда единственный надежный метод – одноразовая лента (аналог одноразового блокнота).

Рандомизированный подход: увеличение числа битов, с которыми необходимо работать криптоаналитику (не увеличивая при этом ключ). Этого можно достичь путем использования больших случайных общедоступных строк. Ключ будет обозначать, какие части этих строк необходимо использовать для шифровки/дешифровки. Тогда криптоаналитику придется использовать метод грубой силы на случайных строках. Стойкость этого метода может быть выражена в терминах среднего числа битов, которые придется изучить криптоаналитику, прежде чем шансы определить ключ станут выше простого угадывания.

Вероятность взлома такого алгоритма зависит от объема памяти, доступного криптоаналитику (но не зависит от его вычислительных ресурсов). Чтобы эта схема стала практичной, требуется около 100 битовых последовательностей по 20 10 бит каждая.

Случайные числа играют важную роль при использовании криптографии в различных сетевых приложениях, относящихся к безопасности. Так, например, использование в лотереях, генерациях паролей, индентификационных номеров и прочее. В криптографии важное значение несут при потоковом шифровании и других различных ответвлений защиты данных.