Курсовой проект

Поточные системы шифрования

Автор: Сорокин Павел Александрович (4 курс, компьютерная безопасность)

Научный руководитель: Просвирнина Ирина Борисовна

Гродно, 2015
г.

Цель курсового проекта

Познакомиться с конструкциями, программной реализацией и направлениями использования поточных систем шифрования.

Задачи курсового проекта

Для достижения поставленной цели необходимо было решить следующие задачи:

- Исследовать предметную область проекта.
- Провести информационный поиск по тебе курсового проекта
- Выявить различия между поточными и блочными шифрами
- Рассмотреть программную реализацию по теме поточных систем шифрования.
- Рассмотреть направления использования поточных систем шифрования.

Введение

Проблема защиты информации появилась с давних времён. Для защиты хранимой и передаваемой информации были придуманы методы обеспечения безопасности информации. К таким методам относится метод шифрования информации.
В современном мире, актуальность данной темы возрастает с каждым годом. С развитием человечества появлялись новые методы и способы шифрования, создавались различные приспособления и машины для шифрования. В наши дни шифрование используется повсеместно и разработана большая база подходов к защите информации посредствам шифрования.

Определение поточных систем шифрования

Поточные шифры относятся к шифрам замены, преобразующим посимвольно открытый текст в шифрованный в зависимости не только от ключа, но и от его расположения в потоке открытого текста.
Изначально шифры замены были построены на основе поточного шифрования и использовали буквы и биграммы. С началось использования электронных систем в шифровании стали использоваться биты и байты.

Основные части структуры поточного шифра

- Управляющий блок(генератор гаммы) – моделируется криптографическим генератором.
- Шифрующий блок – моделируется автоматом Мили с постоянной памятью.

Управляющий блок

Управляющий блок(генератор гаммы) - предназначен генерировать управляющую последовательность, которую используют для формирования шифрующих отображений.

Шифрующий блок

Шифрующий блок - использует данные сформированные управляющим блоком и шифрует символы открытого текста x_i в символы шифрованного текста y_i используя отображения φ_i.

Классификация поточных шифров и их особенности.

Поточные шифры классифицируются на:

- синхронные
- асинхронные (самосинхронизирующиеся)

Синхронные поточные шифры – шифры, в которых поток ключей генерируется независимо от открытого текста и шифротекста.

Самосинхронизирующиеся поточные шифры (асинхронные поточные шифры) – шифры, в которых ключевой поток создаётся функцией ключа и фиксированного числа знаков шифртекста.

Особенности синхронных поточных шифров.

Плюсы синхронных поточных шифров:

- отсутствие эффекта распространения ошибок (только искажённый бит будет расшифрован неверно);
- предохраняют от любых вставок и удалений шифротекста, так как они приведут к потере синхронизации и будут обнаружены.

Минусы синхронных поточных шифров:

- уязвимы к изменению отдельных бит шифрованного текста. Если злоумышленнику известен открытый текст, он может изменить эти биты так, чтобы они расшифровывались, как ему надо.

Особенности асинхронных поточных шифров.

Плюсы асинхронных поточных шифров:

- так как каждый знак открытого текста влияет на следующий шифротекст, статистические свойства открытого текста распространяются на весь шифротекст. Следовательно, асинхронные поточные шифры могут быть более устойчивыми к атакам на основе избыточности открытого текста, чем синхронные поточные шифры.

Минусы асинхронных поточных шифров:

- распространение ошибки (каждому неправильному биту шифротекста соответствуют несколько ошибок в открытом тексте);
- чувствительны к вскрытию повторной передачей.

Критерии оценки криптографических свойств управляющего и шифрующего блоков.

Криптографические свойства поточного шифра определяются свойствами как шифрующего, так и управляющего блока.

К требованиям для управляющей гаммы данного типа шифров относятся:
- управляющая гамма должна иметь большой период, во много раз превосходящий длины шифруемого сообщения, и не должна содержать длинных повторяющихся отрезков.
- управляющая гамма должна иметь большую линейную сложность, чтобы по достаточно длинному отрезку гаммы нельзя было восстановить её полностью за адекватный промежуток времени.
- система, связывающая элементы ключа с известными знаками гаммы, должна быть настолько сложной, что исключается возможность практической реализации алгоритма.

Различия между блочными и поточными шифрами

Заметим, что граница между поточными и блочными шифрами весьма условна и существуют шифры со свойствами присущими как для блочных, так и для поточных шифров.

Блочные шифры:

- разбивают исходное сообщение на блоки определённой длины и обрабатывают их
- одна ошибка влечёт за собой несколько ошибок
- меньшая скорость работы

Поточные шифры:

- обрабатывают весь входящий текст посимвольно
- структура поточного ключа имеет уязвимые места
- в синхронных поточных шифрах отсутствует эффект размножения ошибок
- высокая скорость работы

Направления использования поточных систем шифрования.

Ввиду особенностей свойств поточных шифров, данные системы нашли свою определённую нишу в обеспечении защиты передаваемых данных, где нужна высокая скорость работы шифрующих систем и хорошая криптографическая стойкость.

Основным направлением развития поточных шифров являются сети передачи данных. Также поточные шифры используются в смарт-картах, RFID-метках. Это обусловлено быстрой скоростью обработки данных и отсутствием эффекта размножения ошибок, что в данной сфере является особенно актуальным.

Примеры поточных шифров:

- A5 используется в системах GSM для защиты связи между абонентом и базовой станцией.
- RC4(Rivest cipher 4) – поточный шифр с переменной длиной ключа. Реализован в десятках коммерческих криптопродуктов, например, Lotus Notes, Apple Computers AOCE, Oracle Secure SQL, является частью спецификации стандарта сотовой связи CDPD (Cellular Digital Packet Data).
- Chameleon – поточный шифр, одновременно сочетающий в своей реализации высокую криптостойкость и необычное для надежного шифра свойство, благодаря которому незначительные изменения в ключе вызывают лишь незначительные изменения в гамме.
- Leviathan – поточный шифр, разработанный в кампании Cisco Systems и ориентированный на сетевые приложения.
- WAKE (Word Auto Key Encryption) – асинхронный поточный шифр. WAKE реализован в антивирусном пакете программ Dr. Solomons Anti-Virus.

Программная реализация генератора с внутренней обратной связью на основании линейной функции

```
5⊖    public static void main(String[] args) {
6
7         boolean[] b = {true,false,false,false,true};
8         int T = 30;
9         int N = b.length;
10
11
12
13        for (int t = 0; t < T; t++) {
14
15            for (int i = N-1; i > 0; i--)
16                b[i] = b[i-1];
17            boolean next = (b[N-1] ^ b[N-2]);
18
19            b[0] = next;
20
21            if (b[N-1]) System.out.print("1");
22            else        System.out.print("0");
23        }
24        System.out.println();
25    }
26
27 }
```

В данном примере рассматривается расчёт $b[4] = b[0] \oplus b[1]$

Текущее значение	b[4]	b[3]	b[2]	b[1]	b[0]	K_i
Начальное значение	1	0	0	0	1	
1	0	1	0	0	0	1
2	0	0	1	0	0	0
3	1	0	0	1	0	0
4	1	1	0	0	1	0
5	0	1	1	0	0	1
6	1	0	1	1	0	0
7	0	1	0	1	1	0
8	1	0	1	0	1	1
9	1	1	0	1	0	1
10	1	1	1	0	1	0
11	1	1	1	1	0	1
12	0	1	1	1	1	0
13	0	0	1	1	1	1
14	0	0	0	1	1	1
15	1	0	0	0	1	1
16	0	1	0	0	0	1

Получили последовательность 1000100110101111.

Период равен $2^n-1 = 15$, где n – длина ключа для генератора.

Программная реализация генератора с внутренней обратной связью на основании нелинейной функции

```
5⊖    public static void main(String[] args) {
6
7         boolean[] b = {true,false,false,false,true};
8         int T = 17;
9         int N = b.length;
10
11
12
13        for (int t = 0; t < T; t++) {
14
15            for (int i = N-1; i > 0; i--)
16                b[i] = b[i-1];
17            boolean next = ((b[N-2] ^ !b[N-1])|(b[N-4] ^ b[N-3]));
18
19            b[0] = next;
20
21            if (b[N-1]) System.out.print("1");
22            else        System.out.print("0");
23        }
24        System.out.println();
25    }
26
27 }
```

Текущее значение	b[4]	b[3]	b[2]	b[1]	b[0]	K_i
Начальное значение	1	0	0	0	1	
1	1	1	0	0	0	0
2	1	1	1	0	0	0
3	1	1	1	1	0	0
4	1	1	0	1	1	1
5	1	1	1	0	1	1
6	0	0	1	1	0	1
7	1	1	0	1	1	0
8	1	1	1	0	1	1
9	0	0	1	1	0	1
10	1	1	0	1	1	0
11	1	1	1	0	1	1
12	0	0	1	1	0	1
13	1	1	0	1	1	0
14	1	1	1	0	1	1
15	0	0	1	1	1	1
16	1	1	0	1	1	0

Получили последовательность 0001110110110110.

Период равен 3.

В данном примере рассматривается расчёт $b[4] = (!b[0] \oplus b[1]) \mid (b[2] \oplus b[3])$

Вывод из сравнения нелинейного и линейного регистра сдвига обратной связи

- линейный генератор имеет более простую реализацию и математическое обоснование периода.
- если выбрать подходящую функцию для нелинейного генератора, то получим более криптографически стойкий алгоритм генерации псевдослучайных последовательностей.
- нелинейный регистр сдвига с обратной связью не имеет общего характера, поскольку нет математического обоснования, как получить такой регистр с максимальным периодом.

Заметим, что решением проблемы периодичности генерируемой последовательности может быть применение линейного регистра сдвига с обратной связью с максимальным периодом и затем комбинирование его обратной связи с помощью нелинейной функции.

Заключение

В данном курсовом проекте была рассмотрена тема поточных систем шифрования.
Был проведён анализ предметной области темы. Рассмотрели классификацию поточных систем шифрования. Выявили различия между блочными и поточными шифрами. Изучили направления использования поточных шифров. Реализовали и проанализировали данные из практической части.
Цель курсового проекта полностью достигнута.

Спасибо за внимание!