Принципы синтеза перспективного алгоритма блочного шифрования с длиной блока 128 бит

Василий Шишкин

«РусКрипто'2013»

28 марта, 2013

Содержание доклада

Перейдем к разделу:

Длина блока

Длина блока 64 бита (ГОСТ 28147-89)

- Эффективно для платформ с ограниченными ресурсами
- Ограничения на объем обрабатываемой информации на одном ключе

Длина блока 128 бит

- Эффективная программная реализация
- В большинстве режимов шифрования позволяет обрабатывать на одном ключе до $\mathbf{2^{28}}$ ТБайт информации

Теоретическая стойкость ГОСТ 28147-89

В 2010 - 2011 годах появился ряд работ, в которых получены результаты, снижающие теоретическую стойкость алгоритма ГОСТ 28147-89

- Конструкция алгоритма ГОСТ 28147-89 с теоретической точки зрения не является идеальной, что уже сказывается на его международном статусе стойкого шифра
- Методы, использующие связанные ключи: вероятность применения на практике крайне мала.
  Вместе с тем, предложенные методы следует учитывать при анализе и синтезе алгоритмов выработки сеансовых ключей шифрования

Перейдем к разделу:

Основные принципы синтеза

- Использовать хорошо изученные конструкции и преобразования
- Учесть современные тенденции в области синтеза и анализа блочных шифров
- Не должны быть эффективно применимы известные методы анализа блочных шифров
- Обеспечить достаточный «запас стойкости»
- Перспективный блочный шифр должен допускать эффективные программные и аппаратные реализации для большинства современных платформ
- Длина блока: 128 бит
- Длина ключа: 256 бит

Базовая конструкция

- Только хорошо изученные конструкции
- Эффективность реализации при обеспечении заданного уровня криптографической стойкости

Схема Фейстеля

- высокая степень изученности
- один и тот же алгоритм для за/рас-шифрования

XSL-схема (SP-сеть)

- высокая степень изученности
- меньше итераций – выше скорость

Базовая конструкция

- Только хорошо изученные конструкции
- Эффективность реализации при обеспечении заданного уровня криптографической стойкости

Схема Фейстеля

- высокая степень изученности
- один и тот же алгоритм для за/рас-шифрования

XSL-схема (SP-сеть)

- высокая степень изученности
- меньше итераций – выше скорость

Выбор нелинейного преобразования

Преобразование пространства V_4

- позволяет получать эффективные аппаратные реализации
- обеспечивает «медленное нарастание» нелинейности

Преобразование пространства V_8

- позволяет получать эффективные программные реализации
- лучшие значения основных криптографических характеристик

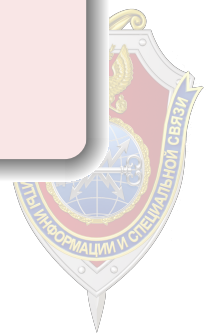
Выбор нелинейного преобразования

Преобразование пространства V_4

- позволяет получать эффективные аппаратные реализации
- обеспечивает «медленное нарастание» нелинейности

Преобразование пространства V_8

- позволяет получать эффективные программные реализации
- лучшие значения основных криптографических характеристик



Синтез нелинейного преобразования

Выбор из известных классов

- близкие к оптимальным значения некоторых криптографических параметров
- очевидная аналитическая структура
- обращение элемента в поле

Случайный поиск с заданным ограничением на параметры

- не оптимальны по совокупности значений основных криптографических характеристик
- не обладают выраженным аналитическим строением

Синтез нелинейного преобразования

Выбор из известных классов

- близкие к оптимальным значения некоторых криптографических параметров
- очевидная аналитическая структура
- обращение элемента в поле

Случайный поиск с заданным ограничением на параметры

- не оптимальны по совокупности значений основных криптографических характеристик
- не обладают выраженным аналитическим строением

Синтез нелинейного преобразования

Эксплуатационные характеристики

- минимизировать число бинарных битовых операций, необходимых для реализации подстановки
- позволяет достичь высокой эффективности аппаратной реализации, а также программной реализации при использовании инструкций процессора, осуществляющих векторную обработку данных (SIMD)

Синтез линейного преобразования

- С криптографической точки зрения: максимизировать показатели рассеивания. Использование максимально рассеивающих (MDS) линейных преобразований
- Высокая скорость реализации
- Объем необходимой памяти – ограничен
- Линейное преобразование $\boldsymbol{V_n \rightarrow V_n}$ эффективно реализуется при $\boldsymbol{n \leq 64}$.

Синтез линейного преобразования

Решение (А.А. Нечаев):

- построение максимально рассеивающего преобразования на основе рекурсивного МДР-кода
- объем необходимой памяти: ms бит вместо $m^2 s$ (16 байт вместо при $s = 8$ и $n = 128$)
- Позволяет разработать программную реализацию, осуществляющую шифрование с высокой скоростью

Линейное преобразование: реализация

Использование памяти при реализации

- использование предвычисленных значений, хранящихся в памяти
- используется при разработке эффективной программной реализации практических всех блочных шифров

Модифицированный метод

- формируются массивы предвычисленных данных двух типов
- больше операций, выше скорость
- модификация метода требует меньший объем памяти для хранения предвычисленных массивов данных

Линейное преобразование: реализация

Использование памяти при реализации

- использование предвычисленных значений, хранящихся в памяти
- используется при разработке эффективной программной реализации практических всех блочных шифров

Модифицированный метод

- формируются массивы предвычисленных данных двух типов
- больше операций, выше скорость
- модификация метода требует меньший объем памяти для хранения предвычисленных массивов данных

Число итераций

- Стойкость блочного шифра относительно многих методов анализа существенно зависит от числа итераций
- Для перспективного алгоритма число итераций должно быть выбрано с запасом, то есть бóльшим, чем необходимо для обеспечения стойкости относительно известных методов

Число итераций

- Стойкость блочного шифра относительно многих методов анализа существенно зависит от числа итераций
- Для перспективного алгоритма число итераций должно быть выбрано с запасом, то есть бóльшим, чем необходимо для обеспечения стойкости относительно известных методов

Алгоритм развертки ключа

- Общие принципы синтеза алгоритмов развертки ключа на данный момент не сформировались
- Алгоритм развертки ключа не должен допускать эффективную применимость методов анализа, использующих связанные ключи
  - криптографически качественные функции выработки итерационных ключей
- Алгоритм развертки ключа не должен существенно снижать эксплуатационные качества шифра
  - использование тех же преобразований, что участвуют в процессе шифрования
  - для записи используемых констант требуется минимальный объем памяти

Перейдем к разделу:

Базовый алгоритм шифрования

- XSL-схема: 9 полных и 1 усеченная итерации
- запас стойкости
- число итераций в 1,5 раза меньше числа итераций AES-256

$$E_{K_1,\ldots,K_{10}} = X[K_{10}]LSX[K_9]\ldots LSX[K_2]LSX[K_1],$$

$$K_i \in V_{128},\ i = 1,\ldots,10$$

Алгоритм развертки ключа

- использование схемы Фейстеля: 32 итерации
- итерационное преобразование совпадает с итерацией шифрования (вместо ключа – константы)
- используемые константы задаются номером итерации схемы Фейстеля

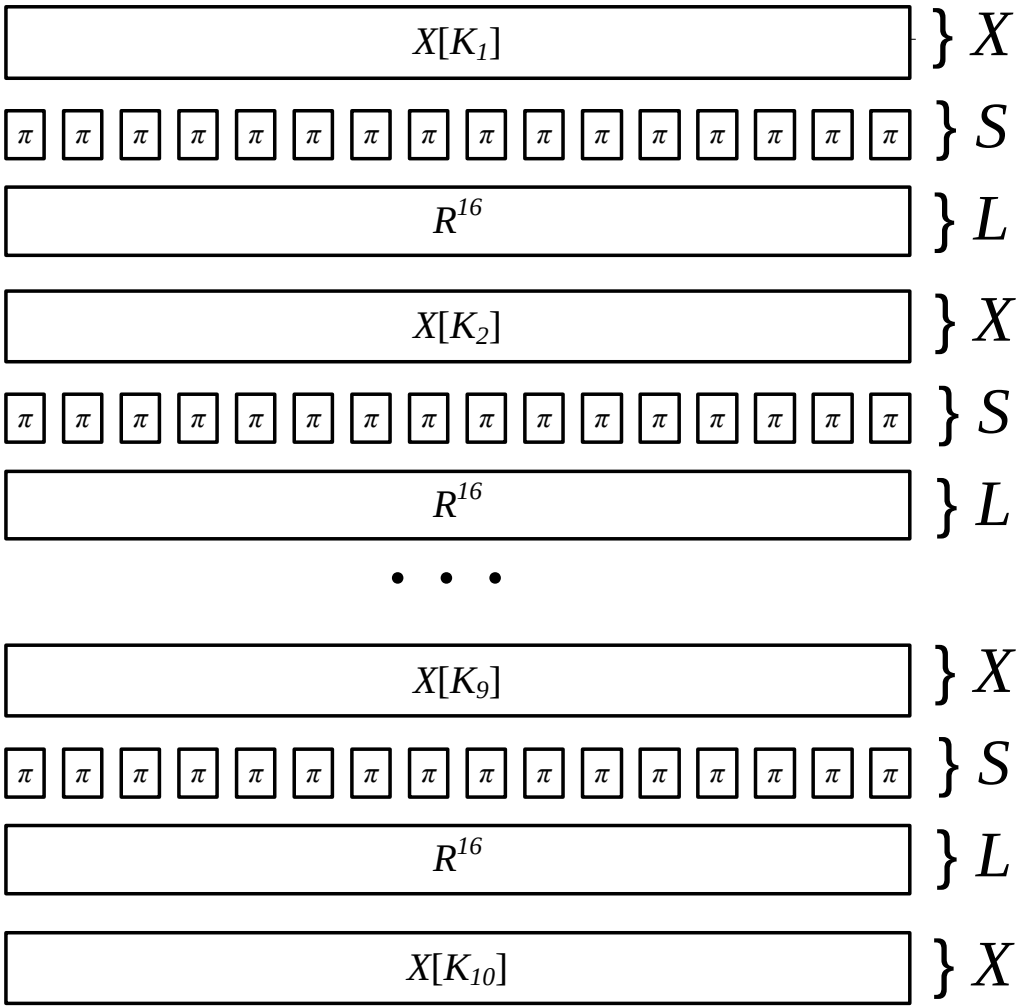
$$K \in V_{256},\ (K_1, K_2) = K,$$

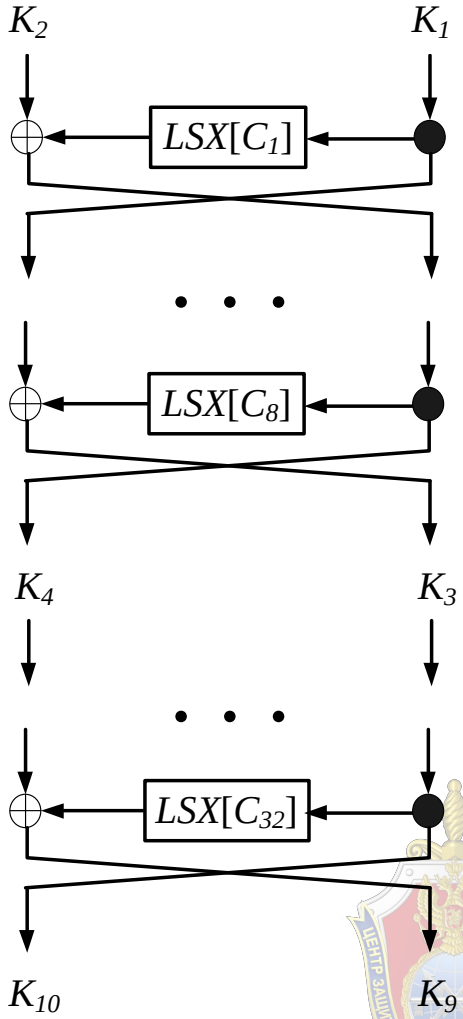
$$(K_{2i+1}, K_{2i+2}) = F[C_{8(i-1)+8}] \dots F[C_{8(i-1)+1}](K_{2i-1}, K_{2i}),$$

$$i = 1, \dots, 4,$$

$$F[C] : V_{128}^2 \to V_{128}^2, \quad F[C](a_1, a_0) = (LSX[C](a_1) \oplus a_0, a_1),$$

$$C_i \in V_{128},\ C_i = L([i]_2), \quad i = 1, \dots, 32$$


X[K1]
X
π π π π π π π π π π π π π π π π
S
R16
L
X[K2]
X
π π π π π π π π π π π π π π π π
S
R16
L
X[K9]
X
π π π π π π π π π π π π π π π π
S
R16
L
X[K10]
X


K2
K1
LSX[C1]
LSX[C8]
K4
K3
LSX[C32]
K10
K9

Нелинейное преобразование

- блок одинаковых фиксированных подстановок
- подстановки множества V_8

$$S(a) = S(a_{15}\| \dots \|a_0) = \pi(a_{15})\| \dots \|\pi(a_0),$$

где $a = a_{15}\| \dots \|a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, \dots, 15$

$$\pi : V_8 \to V_8$$

Линейное преобразование

- реализуется линейным регистром сдвига
- многочлен, задающий закон рекурсии, содержит минимально возможное число единиц в двоичном представлении коэффициентов

$$L : V_{128} \to V_{128}, \quad L(a) = R^{16}(a)$$

$$R : V_{128} \to V_{128},$$

$$R(a) = R(a_{15} \| \dots \| a_0) = l(a_{15}, \dots, a_0) \| a_{15} \| \dots \| a_1,$$

где $a = a_{15} \| \dots \| a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, \dots, 15$
$l : V_8^{16} \to V_8$ – линейное, задается 16-ю умножениями на константы в поле $GF(2^8)$

Предварительные исследования эксплуатационных характеристик

- скорость шифрования на 32-битной платформе сравнима со скоростью шифрования с помощью алгоритма ГОСТ 28147-89
- скорость шифрования на 64-битной платформе превосходит скорость шифрования с помощью алгоритма ГОСТ 28147-89 более чем в 1,4 раза

- усовершенствование программной реализации
- увеличение возможностей аппаратной базы

Предварительные исследования эксплуатационных характеристик

- скорость шифрования на 32-битной платформе сравнима со скоростью шифрования с помощью алгоритма ГОСТ 28147-89
- скорость шифрования на 64-битной платформе превосходит скорость шифрования с помощью алгоритма ГОСТ 28147-89 более чем в 1,4 раза

- усовершенствование программной реализации
- увеличение возможностей аппаратной базы

Спасибо за внимание

Вопросы?