Защита информации

Павел Юдаев

МГТУ им. Баумана, Кафедра ИУ-9

Москва, 2014

Раздел 3 - Блочные шифры. DES

Блочный симметричный шифр

Шифрование:

- Текст $\rightarrow$ набор блоков фиксированного размера.
- Текст $\rightarrow$ дополнение конца текста до длины блока.
- Работа поблочно, длина блока не меняется.
- Алгоритм шифрования может быть рандомизированным.

Расшифрование:

- Работа поблочно, длина блока не меняется.
- Алгоритм расшифрования - всегда детерминированный.

Шифрование одного блока детерминированное.

n - длина блока.
$K = \{0,1\}^q,\ M = \{0,1\}^n,\ C = \{0,1\}^n$

$E : K \times M \to C.$

$D : K \times C \to M.$

$\forall k \in K\ D(k, E(k, m)) = m.$

Скорость работы поточных и блочных шифров

(AMD Opteron 2.2 GHz, ОС Linux, библиотека crypto++ 5.6.0)

Шифр	Размер блока/ключа	Скорость, МБ/с
поточные		
RC4 (1987)		126
Salsa 20/12 (2005)		643
блочные		
3DES (1977, 1997)	64/168	13
AES-128 (2000)	128/128	109

Схема Фейстеля.

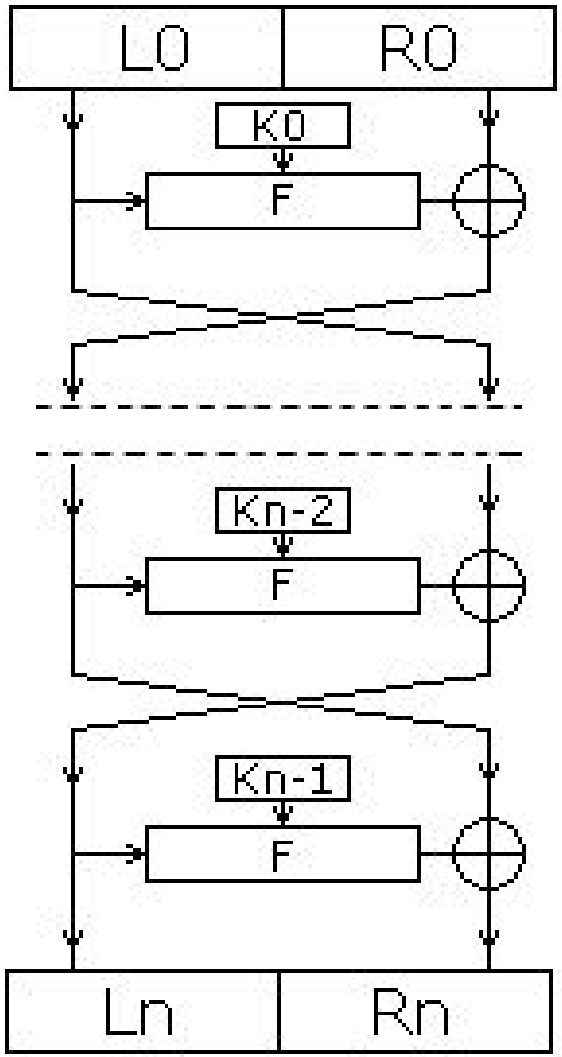
- В 1973 году Хорст Фейстель (Horst Feistel) опубликовал статью “Cryptography and Computer Privacy”, в которой ввел конструкцию, названную впоследствии сетью (схемой) Фейстеля
- Эта схема была использована в проекте Lucifer фирмы IBM, над которым работали Фейстель и Дон Копперсмит (Don Coppersmith)
- Этот проект был экспериментальным, но он стал основой для шифра Data Encryption Standard (DES).

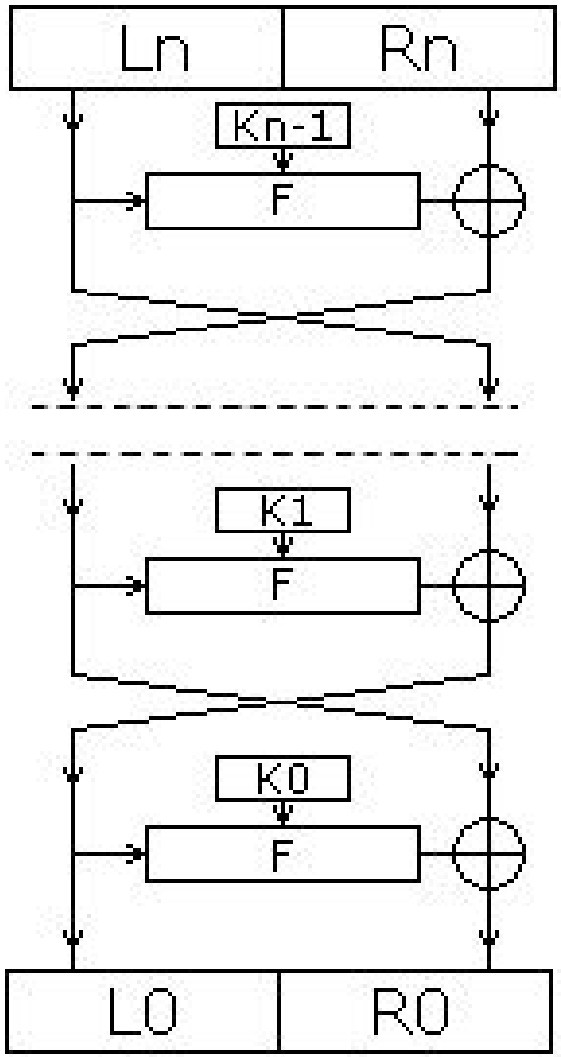
Операция $\oplus$ - сложение по модулю два,
$l \in \{0,1\}^{n/2}$ - левая половина блока,
$r \in \{0,1\}^{n/2}$ - правая,
k_i - ключ для данного раунда, генерируется по ключу k.

Опр.

раунд в схеме Фейстеля с использованием функции
$f(k_i, \cdot) : \{0,1\}^{n/2} \to \{0,1\}^{n/2}$ - это отображение

$H_f : (\{0,1\}^{n/2}, \{0,1\}^{n/2}) \to (\{0,1\}^{n/2}, \{0,1\}^{n/2}) :$
$H_f(l, r) = (r \oplus f_{k_i}(l), l)$


L0
R0
K0
F
Kn-2
F
Kn-1
F
Ln
Rn


Ln
Rn
Kn-1
F
K1
F
K0
F
L0
R0

- На последнем шаге в схеме Фейстеля левая и правая половины блока *не меняются местами*. Поэтому один и тот же алгоритм можно использовать для зашифрования и расшифрования.
- Раундовые функции f_i - произвольные.
- Схема Фейстеля реализует обратимую функцию.
- Схема Фейстеля - общий метод построения обратимой функции из нескольких необратимых. В прошлом она использовалась для построения многих шифров. Но современный шифр AES ее не использует.

Задача

Построить обратное преобразование для одного раунда схемы Фейстеля.

Опр.

$Perms(X)$ - множество всех обратимых функций (перестановок) $X \to X$

Опр.

Функция $f : K \times X \to X$ - это *псевдослучайная перестановка* [по второму аргументу], если
$\forall k\ f(k, \cdot) \in Perms(X)$,
$f \in \mathrm{PT}$, $f^{-1}(k, \cdot) \in \mathrm{PT}$,
$\forall$ оракула $A \in \mathrm{PPT}$ при $k \xleftarrow{R} K$, $r \xleftarrow{R} Perms(X)$
величина
$Adv(A, f) = |P[A(f(k, \cdot)) = 1] - P[A(r(\cdot)) = 1]| < \varepsilon(n)$,

Т.е. оракул анализирует перестановку и выдает ответ: это перестановка из семейства ПСП или случайно выбранная перестановка.
Вероятность того, что он правильно отличит перестановку из семейства ПСП от СП, не более $1/2 + \varepsilon(n)$.
Т.е. преимущество оракула над угадыванием - не более $\varepsilon(n)$.

Теорема 1 (Люби - Ракофф, Luby - Rackoff, 1985)

Пусть $f : \{0,1\}^q \times \{0,1\}^{n/2} \to \{0,1\}^{n/2}$ - псевдослучайная функция и раундовые ключи - значения независимых р.р. с.в.
Тогда 3-раундовая схема Фейстеля
$H : \{0,1\}^{3q} \times \{0,1\}^n \to \{0,1\}^n$,
$H : ((k_1, k_2, k_3), x) \to H_{f_{k_3}}(H_{f_{k_2}}(H_{f_{k_1}}(x)))$
реализует псевдослучайную перестановку.

Без доказательства.

(k_i - незав. р.р. с.в, т.е. раундовые функции выбираются из $\{f(k, \cdot)\}$ независимо и р.р.)

Раздел 3 - Блочные шифры. DES

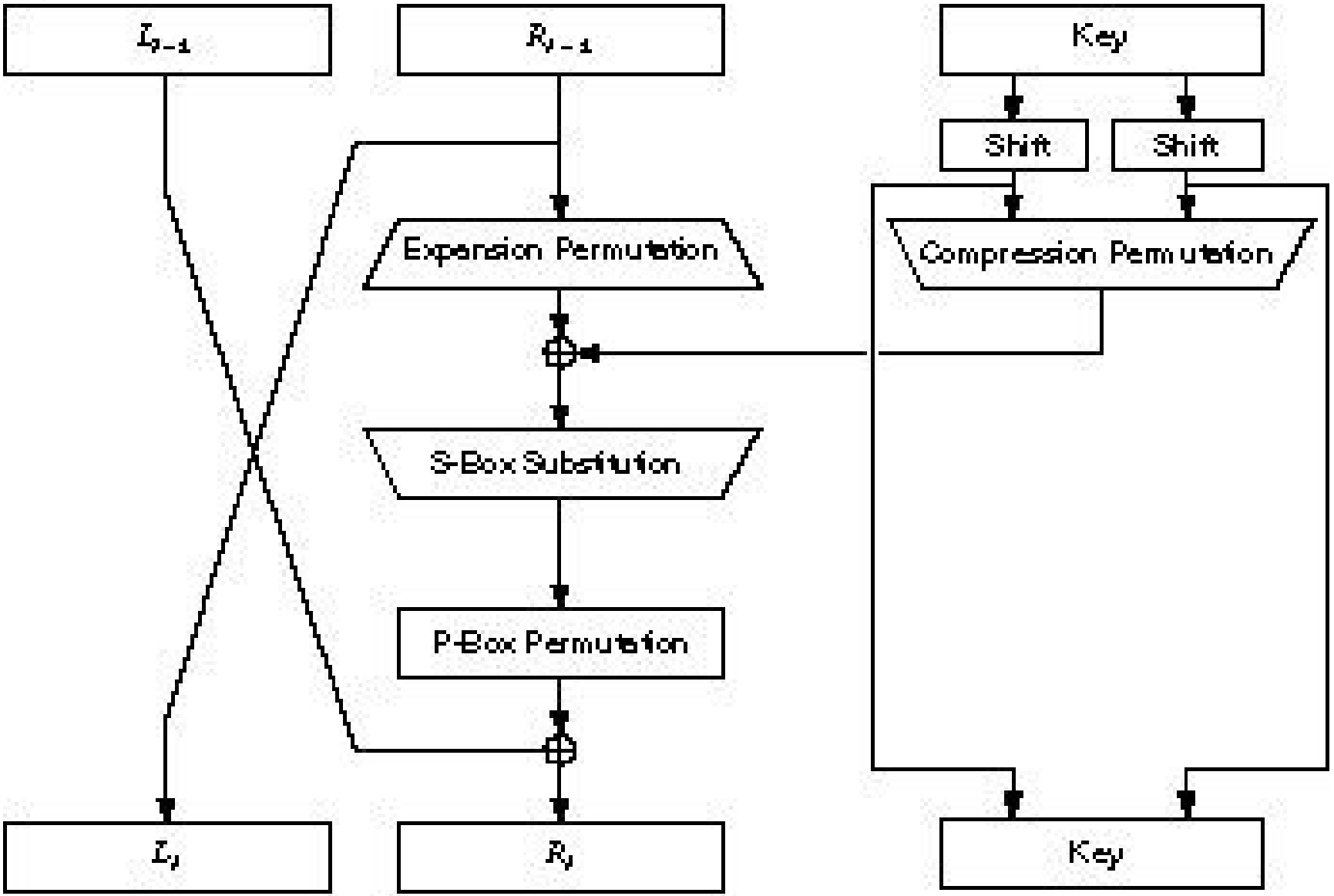
- Разработан фирмой IBM и утвержден правительством США в 1977 году как официальный стандарт шифрования (FIPS-46-3)
- Длина блока - 64 бит.
- Длина ключа 56 бит. Каждый восьмой бит ключа - бит четности, итого 64 бита.
- Алгоритм использует комбинирование нелинейного (S-box) и линейных преобразований
- Для шифрования и расшифрования используются одинаковые алгоритм и ключ.
- Состоит из 16 раундов.
- В 1997 году взломан полным перебором ключей (56 бит!). В качестве временного улучшения предложен шифр 3DES.
- В 2000 новый стандарт - шифр AES.

Шифр DES:

- начальная перестановка IP,
- 16 раундов схемы Фейстеля,
- $FP = IP^{-1}$.

Начальная перестановка IP фиксирована, не влияет на криптостойкость алгоритма.

Один раунд алгоритма DES (1 - 15), т.е. раундовая функция f из схемы Фейстеля:



Генерация расписания ключей k_i: (key schedule)
Уберем биты четности, оставим 56 значащих битов ключа.
Обозн. k_0.
Далее в каждом раунде:

- 56-битная строка - значение k_{i-1} - делится на 28-битных половины.
- Каждая из них независимо сдвигается (с переносом) на 1-2 бита. Величины сдвигов заданы в фиксированной таблице.
- По еще одной фиксированной таблице выбираются 48 бит из 56.
- Получили k_i.

В результате подключ каждого раунда состоит из своего подмножества битов ключа.

Слабые ключи: все раундовые ключи равны.

```
01..0101..01
01..01FE..FE
FE..FE01..01
FE..FEFE..FE
```

(биты четности)

Все раундовые функции одинаковы и
$\forall m\ E(k, m) = D(k, m)$.

Раундовая функция f в схеме Фейстеля шифра DES.

1. E - перестановка с расширением.

$\{0,1\}^{32} \to \{0,1\}^{48}$ в соответствии с фиксированной таблицей.

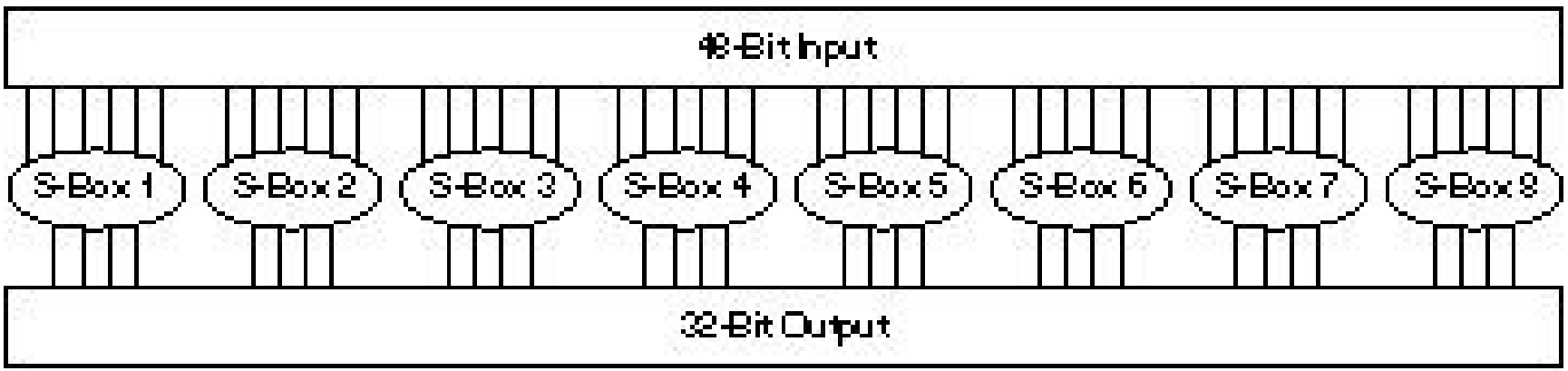
2. Добавление раундового ключа: $x = x \oplus k_i$

3. S - перестановка со сжатием, $\{0,1\}^{48} \to \{0,1\}^{32}$

Набор из 48 бит поступает на вход в 8 S-блоков, каждый $\{0,1\}^6 \to \{0,1\}^4$. S-блоки задаются фиксированными таблицами.

Основное свойство S-блоков - отсутствие линейной зависимости результата от входных данных (см. секцию "Криптоанализ DES").

Почему выбраны числа 6 и 4? Это был максимальный размер, который мог быть реализован в одной микросхеме в начале 70-х.


48-Bit Input
S-Box 1
S-Box 2
S-Box 3
S-Box 4
S-Box 5
S-Box 6
S-Box 7
S-Box 8
32-Bit Output

$$S_i: \{0,1\}^6 \longrightarrow \{0,1\}^4$$

S_5		Middle 4 bits of input															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
Outer bits	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

4. P-блок - фиксированная перестановка, $\{0,1\}^{32} \to \{0,1\}^{32}$.

Цель: результат каждого S-блока в следующем раунде попадет в 4 разных S-блока.

Каждая операция, кроме S-блока, линейная, т.е. может быть описана как $out = B \cdot in$, где B - матрица этой операции.

Цель: по шифротексту одного блока нельзя узнать ничего о блоке открытого текста и о ключе.

Зачем нужен блок E?

$E : \{0,1\}^{32} \to \{0,1\}^{48}$ в соответствии с фиксированной таблицей.

E такой, что после каждого раунда каждый бит результата зависит от все большего числа бит открытого текста. Т.е. достигается *лавинный эффект*.

Конечная цель - чтобы каждый бит шифротекста зависел от каждого бита открытого текста и ключа.

Пусть значение первого бита открытого текста влияет только на первые 8 битов шифротекста.
Пусть $m_1[1] \neq m_2[1],\ m_1[i] = m_2[i]\ \forall i \neq 1$.
Тогда верно $c_1[i] = c_2[i]\ \forall i > 8\ \forall k$.
Т.е. шифр не является псевдослучайной перестановкой.

2. Пусть множество бит шифротекста в позициях $\{j_1, ..., j_l\}$ зависит только от бит открытого текста в позициях $\{i_1, ..., i_l\}$.
Тогда из $c_1[j_1] = c_2[j_1], ..., c_1[j_l] = c_2[j_l]$ следует
$m_1[i_1] = m_2[i_1], ..., m_1[i_l] = m_2[i_l]$.

Сочетание лавинного эффекта и P-блока.

Пусть за один раунд каждый входной бит влияет на два бита результата раунда, причем они находятся в позициях, которые на следующем раунде влияют на другие биты, не те, что уже зависят (идеальный случай).

(Рисунок о влиянии первого бита открытого текста.)

Тогда за i раундов один бит открытого текста повлияет на 2^i битов результата.

Если же j-й бит всегда влияет на j-й и $j+1$-й, то за i раундов один бит открытого текста повлияет на $i+1$ битов результата.

Для “перемешивания” нужен P-блок.

Расшифрование DES

Один и тот же алгоритм, *следовательно, одну и ту же микросхему*, можно использовать для зашифрования и шифрования.

Раундовые ключи:

- Не зависят от данных, вычисляются заранее.
- Порядок при шифровании: $k_1, ..., k_{16}$.
- Порядок при расшифровании: $k_{16}, ..., k_1$.

Реализация DES в UNIX: в библиотеке `crypto(3) - OpenSSL cryptographic library`.

Применение: Первые версии протокола Kerberos (авторизации в гетерогенных компьютерных сетях) использовали шифр DES.

Раздел 3 - Блочные шифры. DES

Шифр DES от NSA:

- S-блоки - набор констант без объяснения того, как они получены и для чего нужны.
- алгоритм сертифицирован NSA (National Security Agency), проблема: наличие лазейки (backdoor), позволяющей NSA легко декодировать сообщения. Наличие лазейки не доказано. Вероятно, ее нет.

Исследовалось,

- почему была выбрана такая длина ключа,
- такое число циклов,
- такой вид S-блоков.

Обнаружили:

- число циклов - достаточное для распространения лавинного эффекта.
- такой выбор S-блоков обеспечивает гораздо более высокий уровень стойкости шифра, чем (в среднем) обеспечил бы случайный выбор S-блоков.

Модель атак на шифр DES: детерминированное шифрование, многоразовый ключ.
Цель: найти ключ быстрее, чем полным перебором.

Опр. (Атака с выбором открытого текста)

Если злоумышленник может выбирать открытые тексты один за другим, получать их шифротексты, и выбирает следующий открытый текст m, зная все предыдущие пары (c, m), то он осуществляет *атаку с адаптивным выбором открытого текста*.

Линейный криптоанализ

Атака с неадаптивным выбором открытого текста.
Known plaintext attack.

Пусть для любых случайных k, m существуют такие подмножества индексов, что

$$\left(\bigoplus_{i\in\{1..64\}} m[i]\right) \oplus \left(\bigoplus_{i\in\{1..64\}} c[i]\right) = \left(\bigoplus_{i\in\{1..56\}} k[i]\right)$$

с вер-ю, заметно отличной от $1/2$.

Обозн. $L_2(m) \oplus L_3(c) = L_1(k)$

В шифре их не должно быть.

Пусть верно $P[L_1(k)\ [\oplus 1]\ = L_2(m) + L_3(c)] = 1/2 + \gamma,\ \ \gamma > 0.$

Алгоритм голосования:
Для нескольких сообщений m получим их шифротексты c.
$N = |\{m\}|$
$T = |\{m : L_2(m) + L_3(c) = 0\}|$
Если $T > N/2$, принять $L_1(k) = 0$
иначе $L_1(k) = 1$

Утверждение

Пусть p_s - вероятность того, что алгоритм голосования дает правильный ответ. Она растет при росте N и γ.

Док-во

Пусть $L_1(k) = 0$. T - с.в., сумма независимых с.в. X_i с распределением Бернулли
$P(X_i = 0) = \frac{1}{2} + \gamma$,
$P(X_i = 1) = \frac{1}{2} - \gamma$.

$$E(X_i) = \frac{1}{2} - \gamma, \quad D(X_i) = \frac{1}{4} - \gamma^2$$

$$E(T) = N(\frac{1}{2} - \gamma),$$
$$D(T) = N(\frac{1}{4} - \gamma^2)$$

Док-во (Продолжение)

Решение неправильное, если $T < N/2$.

Вероятность неправильного решения

$$p_f = 1 - p_s \leq P\left[|T - E(T)| \geq N\gamma\right] \leq$$

$$(*) \leq D(T)/(N\gamma)^2 = \frac{1}{N}\left(\frac{1}{4\gamma^2} - 1\right)$$

(*) - неравенство Чебышева (см. теор. вер.)

Ч.т.д.

Замечание

При $N = 2^{43}$, $\gamma = 2^{-21}$ $\quad p_s > \frac{7}{8}$

Что происходит с вер-ю вып-я линейного соотношения при последовательных раундах DES?

$X_i \in \{0,1\}$, $P(X_i = 0) = 1/2 + \varepsilon_i$, независимы.
Легко видеть, что $P(X_1 \oplus X_2 = 0) = 1/2 + 2\varepsilon_1\varepsilon_2$

Лемма (Лемма о накоплении - Piling up lemma)

Пусть $X_i \in \{0,1\}$, $P(X_i = 0) = 1/2 + \varepsilon_i$, $i = 1..n$, независимы.

Тогда $P(X_1 \oplus ... \oplus X_n = 0) = 1/2 + 2^{n-1} \prod_{i=1}^{n} \varepsilon_i$

Док-во

По индукции.

Замечание

Значение леммы: если $P(X_i = 0) = 1/2 + \varepsilon_i$ и $|\varepsilon_i| < 1/2$,
то $|P(X_1 \oplus ... \oplus X_{i-1} = 0) - 1/2| > |P(X_1 \oplus ... \oplus X_i = 0) - 1/2|$,
т.е. добавление X_i к сумме с.в.
приближает распределение суммы с.в. к равномерному распределению.

Лучшее линейное соотн. для **14** раундов DES имеет вер-ть $\frac{1}{2} + 2^{-21}$.

Есть и другие соотн., плюс анализ **2** раундов DES.

Позволяют найти 14 бит ключа. Остальные 42 бита - перебором.

Результат - атака:

- 2^{43} известных пар $(m, c(m))$
- и еще 2^{43} вычислений DES.
- вер-ть успеха 85%

Полный перебор: 2^{56} вычислений DES.

Поэтому S-блоки должны быть такими, что ни один выходной бит не является “близким” к линейной функции входных битов.

Утверждение

Если выбирать S-блоки и P-блоки случайно, шифр DES будет нестойким: существует атака с выбором открытого текста, которая позволяет найти ключ, в среднем, за время $O(2^{24})$.

Без доказательства.

Задача

Вывести $E(X_i)$, $D(X_i)$, $E(T)$, $D(T)$.

Задача

Дан вектор $x \in \{0,1\}^6$ и вектор $y \in \{0,1\}^6$ - перестановка элементов вектора x. Найти матрицу $A \in \mathbb{B}_{6\times 6}$: $Ax = y$.
Все операции проводятся по модулю 2.

Задача

Дан вектор $x \in \{0,1\}^6$ и вектор $y \in \{0,1\}^4$:
$y[1] = x[2] \oplus x[3]$, $y[2] = x[4]$,
$y[3] = x[1] \oplus x[2] \oplus x[5]$, $y[4] = x[6]$.
Найти матрицу $A \in \mathbb{B}_{4\times 6}$: $Ax = y$.
Все операции проводятся по модулю 2.

Задача

В условии предыдущей задачи $y[2] = x[4] \oplus 1$.
Найти A, b : $Ax \oplus b = y$.

Понятие о дифференциальном криптоанализе

Первая публикация: 1990, Eli Biham, Adi Shamir.

Атака с адаптивным выбором открытого текста.
$m_1, m_2 \Rightarrow \Delta m$
$c_1, c_2 \Rightarrow \Delta c$
$\Rightarrow$ предположение о ключе.
Атака требует 2^{47} выбранных открытых текстов и ш/т: $m, c(m)$.

При разработке DES S-блоки проектировались так, чтобы быть наиболее устойчивыми к дифференциальному криптоанализу.
Публикаций не было!

Подробно о дифференциальном криптоанализе: см. [3]

Атака полным перебором значений ключа

Сколько пар $m, c(m)$ достаточно?

Лемма

Пусть DES - идеальный шифр, т.е. множество ключей порождает 2^{56} случайных обратимых функций.
Тогда $\forall\ (m, c)$ с вероятностью более $1 - 1/256 \approx 0.995$ существует не более одного ключа k такого, что $c = DES(k, m)$.

Док-во

$$P[\exists k_1 \neq k :\ DES(k, m) = DES(k_1, m)] \leq$$

$$\leq \sum_{k_1 \in \{0,1\}^{56}} P[DES(k, m) = DES(k_1, m)] \leq 2^{56} \cdot (1/2^{64}) = 1/256.$$

Ч.т.д.

Для двух пар и шифра DES, вероятность того, что существует не более одного ключа k такого, что
$c_1 = DES(k, m_1),\ c_2 = DES(k, m_2)$, равна около $1 - 1/2^{71}$

Поэтому двух пар $(E(k, m_1), m_1), (E(k, m_2), m_2)$ достаточно для нахождения ключа перебором.

DES: длина ключа 56 значащих бит.

1977: перебор невозможен
1998: deep crack machine, 250K$, 3 дня
2006: COPACOBANA(120 FPGAs), 10K$, 7 дней

Ключа длиной 56 бит было не достаточно уже в 1998. Шифр DES еще был стандартом.

Более стойкие шифры на базе DES: 3DES, DESX.

E-блок и P-блок шифра DES линейны, т.е. их можно представить в виде матричной операции $out = B \cdot in$. Если бы S-блок тоже был линеен, то один раунд схемы Фейстеля и весь шифр можно было бы представить в виде аффинной операции $c = A \cdot m \oplus b$.

Тогда шифр был бы *алгебраичен*: $\forall k_1, k_2 \; \exists k_3 : \; E_{k_1} \circ E_{k_2} = E_{k_3}$. Для шифра DES это соотношение не выполняется. Поэтому многократное шифрование с разными ключами делает шифр более сложным для взлома полным перебором.

Атака "встреча посередине"

Пусть $2DES((k_1, k_2), m) = E(k_1, E(k_2, m))$.

Для пары (c, m) найти ключ $k = (k_1, k_2)$ такой, что $c = E(k_1, E(k_2, m))$ эквивалентно задаче найти ключ такой, что $E(k_2, m) = D(k_1, c)$

Таблица соответствия $z = E(k_2, m) \leftrightarrow k_2$ для левой части, сортировать по z. Размер таблицы 2^{56}.

$\forall\, k_1$ выч. $y = D(k_1, c)$, искать его в этой таблице, среди значений z.

Нашли пару (k_1, k_2) : $2DES((k_1, k_2), m) = c$.

Время работы: построение и сортировка таблицы $2^{56} log(2^{56})$, поиск в таблице $2^{56} \cdot log(2^{56})$.

Итого $2^{63} \ll 2^{112}$.

Шифр 3DES

Опр.

$3DES((k_1, k_2, k_3), m) = E(k_1, D(k_2, E(k_3, m)))$

Длина ключа $3 \times 56 = 168$ бит, в 3 раза медленнее DES.

Задача

Доказать, что аналогичная атака на 3DES требует времени $O(2^{112})$, памяти $O(2^{56})$.

Шифр DESX

Опр.

Пусть $E : K \times \{0,1\}^k \to \{0,1\}^n$ - блочный шифр. Определим $EX((k_1, k_2, k_3), m) = k_1 \oplus E(k_2, m \oplus k_3)$.

Задача

Почему DESX стойкий к “встрече посередине”?

Длина ключа шифра DESX равна $64 + 56 + 64 = 184$ бита, но есть простая атака за время $O(2^{64+56} log(2^{64+56}) = O(2^{132})$.

Задача

Что это за атака?

Задача

Конструкции $k_1 \oplus E(k_2, m)$ и $E(k_1, m \oplus k_2)$ не улучшают стойкость шифра. Почему?

Этот слайд - факультативный материал

Взлом шифра перебором на квантовом компьютере

Опр.

Задача поиска в общем виде:
пусть $f : X \to \{0, 1\}$. Задача: найти $x \in X : f(x) = 1$

Классический компьютер: время $O(|X|)$

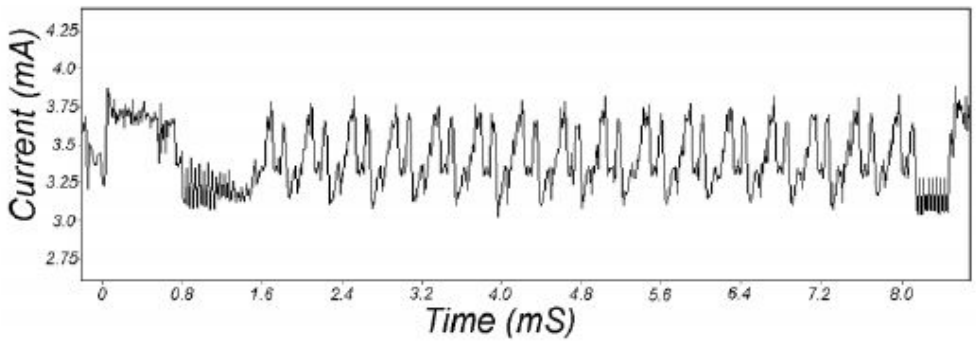
Квантовый компьютер: $O(\sqrt{|X|})$ - алгоритм Гровера (Grover).
Нет подходящего “железа”.

Защита от перебора на к.к.: 256-битный ключ (напр., AES-256)

Атака по побочным каналам (side channel attack)

Получили в свое распоряжение смарт карту.

Измеряем **время** и/или потребляемый **ток**.



[Kocher, Jaffe, Jun, 1998]

Различные операции выполняются за различное время/ток, в зависимости от входных данных. Проводя измерения и их статистический анализ, можно получить полную информацию о ключе.

Атака на основе ошибок при вычислениях

Воздействуем ЭМИ на смарткарту в определенные моменты времени. Вызываем ошибки при вычислениях. Это раскрывает некоторую информацию о ключе.

Шифр ГОСТ:

- ключ - 256 битов
- сеть Фейстеля
- 32 раунда
- S-блоки м.б. секретными.

Некоторые итоги:

A. Стойкость шифров с одноразовым ключом:
A.1. Абсолютная стойкость к атаке с известным шифротекстом.
A.2. Семант. стойкость к атаке с известным шифротекстом.

B. -

C. Шифры:
C.1. Алфавитные.
C.2. Одноразовый блокнот.
C.3. Поточные шифры, ГПСЧ.
C.4. Блочные шифры. Шифр DES и его анализ.

Литература к лекции.

Линейный криптоанализ:

1. P. Junod, *Linear Cryptanalysis of DES, глава 2.*
`http://crypto.junod.info/lincrypt.pdf`

2. M. Matsui, *Linear Cryptanalysis Method for DES Cipher*

Дифференциальный криптоанализ*:

3*. A. Biham, *Differential Cryptanalysis*
`http://www.cs.haifa.ac.il/~orrd/BlockCipherSeminar/Lecture2-Differential.pdf`

Атаки по побочным каналам*:

4*. Paul C. Kocher. Timing attacks on implementations of Diffie-Hellmann, RSA, DSS, and other systems

5*. J.-F. Dhem, F. Koeune, et al. A practical implementation of the timing attack