Атаки на DES методом дифференциального криптоанализа на основе сбоев.

Differential fault analysis, DES.

Ситало Алексей Юрьевич

26 апреля 2007 года

Эссе по курсу "Защита информации", кафедра радиотехники, Московский физико-технический институт (ГУ МФТИ), http://www.re.mipt.ru/infsec

Основоположниками атак на основе ошибок являются три специалиста американской компании Bellcore: Дэн Боне (Dan Boneh), Ричард ДеМилло (Richard DeMillo) и Ричард Липтон (Richard Lipton). В своей работе, вышедшей в 1996 г., они предложили данный вид атак и описали некоторые из возможных атак на основе ошибок на ряд асимметричных криптосистем. Предложенную идею существенно развили и распространили на алгоритмы симметричного шифрования известнейшие израильские криптоаналитики Эли Бихам (Eli Biham) и Эди Шамир (Adi Shamir), которые в своей работе предложили дифференциальный криптоанализ на основе сбоев (differential fault analysis, DFA). Бихам и Шамир в качестве атакуемого алгоритма выбрали DES.

Атаки на основе сбоев (fault attacks) подразумевают различные специфические воздействия на шифратор с целью нарушения его нормального функционирования, в результате чего шифратор может давать сбои в процессе своей работы (см рис. 1). Такие наведенные криптоаналитиком ошибки в работе шифратора могут дать ему существенно больше информации, полезной для дальнейшего анализа.

Рис.1.

Виды активного воздействия на шифратор:

1. Изменение напряжения питания шифратора, существенно превосходящее допустимые пределы (spike attack).
2. Изменение тактовой частоты шифратора, также выходящее за допустимые рамки (glitch attack).
3. Высокоточное облучение шифратора с помощью лазера, ультрафиолетовым,

рентгеновским или каким-либо другим излучением (optical & radiation attacks).

4. Высокоточное наведение электромагнитного поля или локальный нагрев определенной области шифратора (electromagnetic & heating attacks).
5. Внесение изменений в конструкцию шифратора, например, нарушение определенных электрических контактов.

При этом криптоаналитик контролирует следующие факторы :

- местоположение сбоя (например, конкретный бит обрабатываемых данных);
- время возникновения сбоя (например, номер раунда алгоритма шифрования, при выполнении которого происходит сбой);
- количество бит, подверженных сбою;
- вид сбоя: инверсия значения бита или его сброс (в 0 или 1 в зависимости от технологических особенностей шифратора и вида воздействия).

Идеальной мишенью для атак на основе сбоев являются криптографические смарт-карты, которые, фактически, находятся в полном распоряжении их владельца: криптоаналитик, таким образом, может применить к смарт-карте все перечисленные виды воздействий с целью определения прошитого в смарт-карте секретного ключа[1]

Дифференциальный анализ на основе сбоев.

Рассмотрим алгоритм шифрования DES (см рис. 2). Суть атаки на DES методом дифференциального криптоанализа на основе сбоев состоит в следующем:

1. Предполагается, что в процессе шифрования возникает инверсия одного бита в правой части регистра, содержащего текущее значение шифруемого блока данных (т.е. в $\mathbf{B_i}$ – см. рис. 2). Точное расположение бита, в котором возникает сбой, а также номер раунда, в процессе которого бит инвертируется, атакующему не известны.
2. Атакующий «прогоняет» через шифратор один и тот же текст дважды, при этом, воздействуя на шифратор в одном из этих случаев. В результате у криптоаналитика появляется два шифртекста, представляющих собой один и тот же открытый текст, зашифрованный на одном и том же ключе, но только один их этих шифртекстов является корректным, другой же содержит описанную выше ошибку.
3. Сравнивая два полученных шифртекста, атакующий пытается определить номер раунда, в котором возник сбой:

    - если ошибка возникла в раунде 16, то в правой половине шифртекстов будет различаться только 1 бит (здесь и далее не принимается в расчет финальная перестановка), в левой же различаться будут биты, соответствующие выходным битам той таблицы замен, входное значение которой содержит ошибочный бит (таких таблиц может быть и две);
    - ошибка в раунде 15 также достаточно легко определяется: в этом случае различия в правой половине шифртекстов (которые аналогичны различиям в их левой половине в случае, если ошибка возникла в раунде 16) состоят в битах, соответствующих выходным битам одной или двух таблиц замен; проанализировав различия в левой и правой частях шифртекстов, достаточно легко определить, действительно ли ошибка возникла в раунде 15;
    - несколько сложнее, но возможно определить случай, когда ошибка возникает в раунде 14;
    - в случае, если ошибка возникает в более ранних раундах, данная пара шифртекстов является непригодной для анализа и отбрасывается.

4. К полученным шифртекстам применяется технология дифференциального

криптоанализа. Наиболее простой вариант возникает в случае, если сбой возник в 16 раунде: дифференциальный криптоанализ позволяет легко вычислить значения 6 бит ключа последнего раунда, т.е. $\mathbf{K_{16}}$.

5. Для нахождения значений остальных бит $\mathbf{K_{16}}$ аналогичным образом «прогоняются» через шифратор и анализируются дополнительные тексты. Для полного вычисления ключа последнего раунда достаточно от 50 до 200 шифртекстов.
6. Согласно процедуре расширения ключа алгоритма DES, $\mathbf{K_{16}}$ содержит 48 из 56 бит исходного ключа шифрования алгоритма DES. Остальные биты можно легко вычислить перебором возможных 256 вариантов или анализом отброшенных ранее шифртекстов с ошибками на более ранних раундах.

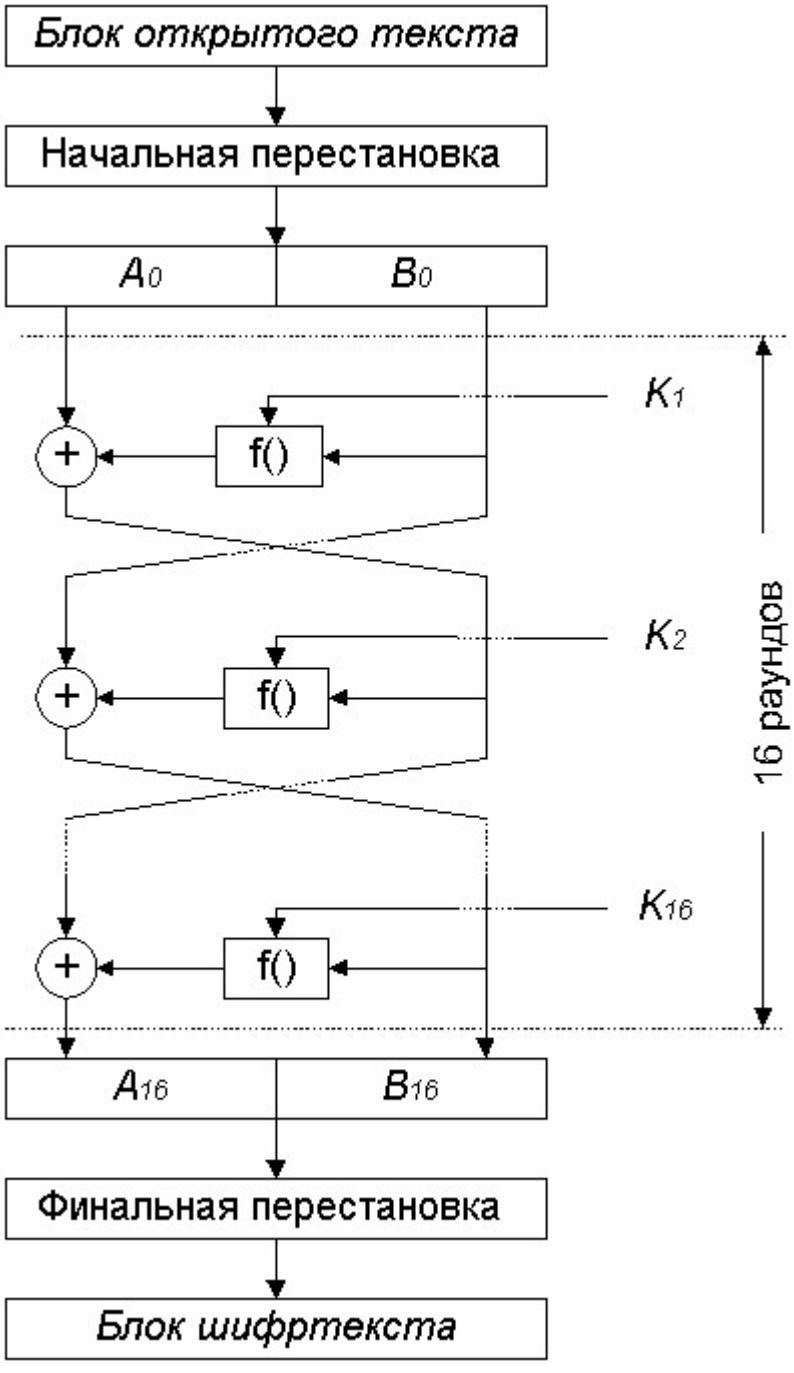

Рис 2. Структура алгоритма DES.

Для данной атаки требуется несравнимо меньше данных, чем для собственно дифференциального криптоанализа алгоритма DES и других алгоритмов. Однако, модель атаки является достаточно строгой и сложно реализуемой на практике. Тем не менее, данный метод криптоанализа является весьма мощным и может быть распространен на другие модели атак.

Дифференциальный метод криптоанализа DES.

Дифференциальный метод криптоанализа (ДКА) был предложен Э.Бихамом и А.Шамиром в 1990 г. Дифференциальный криптоанализ - это попытка вскрытия секретного ключа блочных шифров. При анализе предполагается, что на каждом цикле используется свой подключ шифрования.Данный метод основан на анализе пар открытых текстов, между которыми существует определенная *разность* (difference). При анализе различных алгоритмов разность текстов может быть определена различным образом. Для DES разность открытых текстов $\mathbf{M_1}$ и $\mathbf{M_2}$ определена как операция XOR между данными текстами:

$$\mathbf{D = M_1 \text{ Å } M_2.}$$

Дифференциальный криптоанализ использует множество пар текстов с определенной разностью, анализ которых позволяет выделить некий ключ (или его фрагмент), который является искомым ключом либо однозначно, либо с наибольшей (по сравнению с другими возможными ключами) вероятностью.

Выполняется такой анализ следующим образом. Предположим, имеется два открытых текста, которые на входе в функцию **f()** какого-либо раунда алгоритма имеют разность $\mathbf{D_b}$:

$$\mathbf{D_b = b_1 \text{ Å } b_2.}$$

Разность $\mathbf{D_{be} = be_1 \text{ Å } be_2}$, т.е. разность после обработки $\mathbf{b_1}$ и $\mathbf{b_2}$ расширяющей перестановкой EP, весьма легко определить, поскольку:

$$\mathbf{D_{be} = EP(b_1) \text{ Å } EP(b_2) = EP(b_1 \text{ Å } b_2) = EP(D_b).}$$

Наложение фрагмента ключа операцией XOR вообще не меняет разность, т.е.:

$$\mathbf{D_{bk} = D_{be}.}$$

Аналогично преобразованию EP, легко вычислить разность $\mathbf{D_{bp}}$ после перестановки P:

$$\mathbf{D_{bp} = P(bs_1) \text{ Å } P(bs_2) = P(bs_1 \text{ Å } bs_2) = P(D_{bs}).}$$

Таким образом, единственной операцией из выполняемых функцией **f()**, существенно влияющей на значение разности, остается табличная замена. Значение $\mathbf{D_{bs}}$ зависит не только от разности $\mathbf{D_{bk}}$, но и от конкретных входных значений $\mathbf{bk_1}$ и $\mathbf{bk_2}$. Здесь-то и проявляется влияние наложенного предыдущей операцией ключа шифрования.

Как было сказано выше, таблицы замен меняют 6-битное входное значение на 4-битное. Это означает, что любой входной разности $\mathbf{D_{bk,n}}$ (где **n** – номер таблицы)

соответствует 2^6 = 64 возможных пар входных значений (обозначим их $\mathbf{bk_{1,n}}$ и $\mathbf{bk_{2,n}}$). Соответственно, любой выходной разности $\mathbf{D_{bs,n}}$ соответствует 2^4 = 16 возможных пар $\mathbf{bs_{1,n}}$ и $\mathbf{bs_{2,n}}$. Дифференциальный криптоанализ алгоритма DES эксплуатирует тот факт, что, во-первых, каждое конкретное значение $\mathbf{D_{bk,n}}$ приводит не ко всем возможным 16 значениям $\mathbf{D_{bs,n}}$, а во-вторых, данные значения $\mathbf{D_{bs,n}}$ имеют весьма различную вероятность. Бихам и Шамир в качестве примера рассматривают таблицу S1 и ее входную разность $\mathbf{D_{bk,1}}$ = 34. Возможные значения $\mathbf{D_{bs,1}}$ и их вероятности (в количестве значений пар $\mathbf{bk_{1,1}}$ и $\mathbf{bk_{2,1}}$ из 64 возможных, которые приводят к данному значению $\mathbf{D_{bs,1}}$) приведены в следующей таблице [1]:

Значение $\boldsymbol{D_{bs,1}}$	*Вероятность*
1	8
2	16
3	6
4	2
7	12
8	6
D	8
F	6
Остальные	0
Всего	*64*

Как с помощью всего этого можно определить ключ, рассмотрим на простейшем примере. Предположим, что в распоряжении криптоаналитика имеется пара текстов с $\mathbf{D_{bk,1}}$ = 34. Кроме того, у криптоаналитика есть соответствующие им шифротексты (опять же, предположим, что для их зашифрования использовался *однораундовый* DES) с $\mathbf{D_{bs,1}}$ = 4. Входные значения S1 при таких условиях известны – это значения 13 и 27 ($\mathbf{bk_{1,1}}$ = 13, а $\mathbf{bk_{2,1}}$ = 27, или наоборот). Получается, что криптоаналитик знает значения $\mathbf{be_1}$ и $\mathbf{be_2}$ (поскольку ему известны открытые тексты), а также два варианта значений $\mathbf{bk_{1,1}}$ и $\mathbf{bk_{2,1}}$. В результате криптоаналитик может простейшей операцией XOR вычислить 2 возможных варианта первых шести бит $\mathbf{K_1}$. Выбрать из двух вариантов правильный криптоаналитику поможет вторая пара открытых текстов, если таковая у него есть. Даже если такой пары нет,

криптоаналитик существенно сузил область возможных значений ключа: в 2^5 раз.

Противодействие активным атакам

К сожалению, какого-либо универсального «противоядия» против описанных здесь методов воздействия на шифратор не существует. Однако, существенно усложнить проведение атак на основе сбоев против аппаратного шифратора можно, в частности, следующими способами:

1. Внедрение в шифратор детекторов различных воздействий (напр., детекторов изменения напряжения, частоты питания и/или синхронизации, освещенности и т.д.), которые, при обнаружении воздействия выполняли бы блокировку шифратора.
2. Различного рода пассивное экранирование шифратора, устранение которого приводило бы к выходу шифратора из строя.
3. Различные виды дублирования вычислений со сравнением результатов.

Для программных шифраторов также предлагаются методы защиты:

1. Использование контрольного суммирования фрагментов данных с периодической проверкой в процессе вычислений или различные контрольные вычисления.
2. Дублирование вычислений со сравнением результатов.
3. Внедрение в программу случайных избыточных вычислений.

Ясно, что подобные методы приводят к удорожанию шифратора и/или снижению его быстродействия, однако, видно, что цель в данном случае оправдывает средства.

Использованные источники.

1. «Атаки на алгоритмы шифрования». Панасенко Сергей Петрович. 2007 год. «CIO-World». http://www.cio-world.ru/bsolutions/e-safety/308455/
2. «**Методы криптоанализа классических шифров».** А.Г. Ростовцев, Н.В. Михайлова http://www.ssl.stu.neva.ru/psw/crypto/rostovtsev/cryptoanalysis.html
3. «Современные методы вскрытия алгоритмов шифрования, часть 3». Панасенко Сергей Петрович. 2006 год. «CIO-World». http://www.cio-world.ru/weekly/295841/page2.html