Алгоритм IDEA

Алгоритм IDEA впервые был предложен Ксуеджа Лаем (Xuejia Lai) и Джеймсом Мэсси (James Massey) из швейцарского федерального института технологий в 1990 году. Целью разработки IDEA было создание относительно стойкого криптографического алгоритма с достаточно простой реализацией. Первоначально он назывался PES (Proposed Encryption Standard – Предлагаемый стандарт шифрования), но после демонстрации Бихамом и Шамиром в 1991 году широких возможностей дифференциального криптоанализа шифр был доработан и укреплен против атак такого типа. Название алгоритма сменилось на IPES (Improved Proposed Encryption Standard – Улучшенный предлагаемый стандарт шифрования). Годом позже название IPES сменилось на IDEA (International Data Encryption Algorithm – Международный алгоритм шифрования данных).

До сих пор криптоаналитикам не удалось взломать шифр, хотя были успешние попытки атак вариантов с уменьшенным числом раундов. Считается, что это один из самых лучших и надежных блочных алгоритмов, опубликованных до настоящего времени.

Что касается будущего алгоритма IDEA, то оно пока не совсем ясно. Алгоритм запатентован швейцарской фирмой Ascom и должен быть лицензирован при использовании в коммерческих приложениях. Фирма, правда, разрешила бесплатное некоммерческое использование своего алгоритма (он применяется в общедоступном пакете конфиденциальной версии электронной почты PGP). Частично поэтому попыток заменить им DES не было, а частично потому, что алгоритм достаточно нов, и люди выжидают, наблюдая, как он проявит себя в предстоящие годы криптоанализа.

Обзор алгоритма IDEA

IDEA является одним из нескольких симметричных криптографических алгоритмов, которыми первоначально предполагалось заменить DES.

Как и стандарт DES, IDEA шифрует 64-битовые блоки от крытого текста. IDEA работает с ключами длиной 128 бит. Для зашифрования и расшифрования используется один и тот же алгоритм. Меняется лишь расписание ключей.

Подобно некоторым другим шифрам, алгоритм IDEA использует рассеивание и перемешивание. В основе конструкции алгоритма лежит «смешение операций различных алгебраических групп». Исходный текст в IDEA делится на четыре группы по 16 бит. Для того чтобы комбинировать 16 битные коды, используется три операции:

1. Операция XOR
2. Сложение по модулю 2^{16}
3. Умножение по модулю $2^{16}+1$. (Эту операцию можно рассматривать как S-блок алгоритма IDEA).

Вышеперечисленные операции производятся только с 16-битовыми блоками.

Описание алгоритма IDEA

Рассмотрим подробнее алгоритм IDEA. Исходное сообщение X, имеющее длину 64 бита, делится на четыре 16-битовых подблока X_1, X_2, X_3, X_4. Эти подблоки используются как входы первого раунда алгоритма. Алгоритм состоит из 8 раундов. На каждом из них четыре подблока подвергаются операциям XOR, сложениям и умножениям друг с другом

и шестью 16-битовыми подключами. Между раундами второй и третий подблоки обмениваются местами. Наконец в выходном преобразовании четыре подблока объединяются с четырьмя подключами. На каждом раунде выполняются следующие операции:

1. X_1 умножается на первый подключ.
2. X_2 складыватеся со вторым подключом.
3. X_3 складывается с третьим подключом.
4. Перемножаются X_4 и четвертый подключ.
5. Выполняется операция XOR над результатами шагов 1 и 3.
6. Выполняется операция XOR над результатами шагов 2 и 4.
7. Результаты шага 5 умножаются на пятый подключ.
8. Складываются результаты шагов 6 и 7.
9. Результаты шага 8 умножяются на шестой подключ.
10. Складываются результаты шагов 7 и 9.
11. Выполняется операция XOR над результатами шагов 1 и 9.
12. Выполняется операция XOR над результатами шагов 3 и 9.
13. Выполняется операция XOR над результатами шагов 1 и 10.
14. Выполняется операция XOR над результатами шагов 4 и 10.
15. Результаты шагов 12 и 13 меняются местами (кроме последнего раунда).

На выходе раунда получается четыре подблока, которые служат входом для следующего раунда.

После восьмого раунда выполняется заключительное преобразование:

1. X_1 умножается на первый подключ.
2. Складываются X_2 и второй подключ.
3. Складываются X_3 и третий подключ.
4. X_4 умножается на четвертый подключ.

Конкатенация результатов этих четырех преобразований и есть зашифрованный текст.

В алгоритме используется 52 подключа, т.к. в каждом из восьми раундов используется по шесть ключей и еще четыре необходимо для заключительного преобразования. Эти ключи генернируются следующим образом. Сначала 128-битный ключ разделяется на восемь 16-битных подключей. Это первые восемь подключей алгоритма (шесть из них используются в первом раунде и 2 - во втором). Затем ключ циклически сдвигается налево на 25 битов и снова делится на восемь подключей. Первые четыре подключа используются в раунде 2, а оставшиеся четыре в раунде 3. Ключ циклически сдвигается налево на 25 битов для получения следующих восьми подключей. Этот процесс продолжается до завершения алгоритма.

Расшифрование осуществляется аналогично шифрованию. Порядок использования ключей обращается, а сами они мультипликативно или аддитивно обратны ключам шифрования. При мультипликативном обращении нулевого ключа по модулю $2^{16}+1$ его полагают равным $2^{16}=-1$ и, таким образом, $0^{-1}=0$.

Криптоанализ алгоритма IDEA

Длина ключа IDEA, будучи равной 128 бит, более чем в два раза превышает длину ключа DES. В настоящее время не существует практических методов криптоанализа IDEA, которые были бы эффективнее чем лобовое вскрытие. Но даже при этом условии,

для вскрытия ключа потребуется 2^{128} (10^{38}) шифрований. Как пишет Б. Шнайер: «Создайте микросхему, которая может проверять миллиард ключей в секунду, объедините миллиард таких микросхем, и вам потребуется 10^{13} лет для решения проблемы. Это больше возраста Вселенной. 10^{24} таких микросхем могут найти ключ за день, но во Вселенной не найдется столько атомов кремния, чтобы построить подобную машину».

Алгоритм все еще остается достаточно новым и неисследованным. Возможно, лобовое вскрытие нельзя полагать лучшим методом атаки. IDEA оказался гораздо устойчивей DES к очень удачной дифференциальной криптоатаке Бихама и Шамира, равно как и к линейному криптоанализу. Создатели IDEA макисмально повысили устойчивость алгоритма к дифференциальному криптоанализу. Выдвинув концепцию Марковского шифра, они показали, что устойчивость к дифференциальному криптоанализу можно моделировать и оценить количественно. Известные криптоаналитики потерепели неудачу при попытках взломать IDEA. Лай утверждал (он привел подтверждение, но не доказательство), что алгоритм IDEA устойчив к дифференциальному криптоанализу уже после четвертого из восьми раундов. Согласно Бихаму, его попытка взломать IDEA криптоанализом на основе связанных ключей тоже провалилась.

Исследовав три алгебраические операции IDEA, Вилли Майер показал, что, хотя они и несовместимы, иногда эти операции можно упростить так, чтобы до некоторой степени облегчить криптоанализ. Ему удалось выполнить успешную атаку на 2-раундовый IDEA, и эта атака оказалась эффективнее любого вскрытия (2^{42} операции), но для IDEA с тремя и более раундами эффективность этого вскрытия была ниже лобового вскрытия. Надежность полного 8-раундового алгоритма IDEA осталась непоколебимой.

Как показала Джоан Дэймен, у IDEA имеется целый класс слабых ключей. Однако, эти ключи не являются слабыми в том смысле, в котором слабы некоторые ключи DES, для которых функция шифрования обратна самой себе. Ключ из данного класса может быть легко восстановлен взломщиком при помощи атаки с подобранным открытым текстом. Например, к слабым относится следующий ключ (в шестнадцатеричной записи):

0000, 0000, 0x00, 0000, 0000, 000x, xxxx, x000,

где в позиции «x» может стоять любая цифра.

При использовании такого ключа побитовая операция XOR определенных пар открытых текстов эквивалентна побитовой операции XOR результирующих пар шифртекстов.

Однако, в любом случае вероятность случайной генерации одного из таких слабых ключей крайне незначительна: $\left(\frac{1}{2}\right)^{96}$. На практике вероятность случайного выбора такого ключа можно свести к нулю, модифицировав IDEA так, чтобы исключить появление слабых ключей. Для этого достаточно выполнить операцию XOR каждого подключа с числом 0x0dae.

Отметим, что несмотря на огромное число попыток криптоанализа алгоритма IDEA с полным числом раундов, до сих пор не известна ни одна успешная.

Режимы работы и варианты IDEA

Алгоритм IDEA можно исполнять в любом режиме работы блочного шифра. Подобно DES, любые реализации двойного алгоритма IDEA уязвимы к атаке «встреча посередине». Однако эта атака является непрактичной, т.к. ключ IDEA более чем вдвое длиннее ключа DES. Для того, чтобы осуществить такую атаку, понадобится хранилище данных с размером $64 \cdot 2^{128}$ бит, т.е. 10^{39} байт, что является невозможным в современных условиях.

Можно усилить криптостойкость IDEA, сделав реализацию алгоритма с независимыми подключами, особенно если средство управления ключей позволяют использовать более длинные ключи. Для IDEA нужно всего 52 16-битовых ключа, общей длиной 832 бит. Этот вариант определенно надежнее, правда, никто не может сказать насколько.

Казалось бы, что IDEA можно модифицировать, просто удвоив размер блока, кроме того, при такой модификации алгоритм превосходно работал бы с 32-битовыми подблоками вместо 16-битовых и 256-битовым ключом. Шифрование выполнялось бы быстрее, и стойкость возросла бы в 2^{32} раз. Однако, теория, на которой основан алгоритм, опирается на предположение о простоте числа $2^{16}+1$. Но число $2^{32}+1$ не является простым. Скорее всего, алгоритм и можно изменить так, что бы он работал, но его стойкость будет совсем не та.

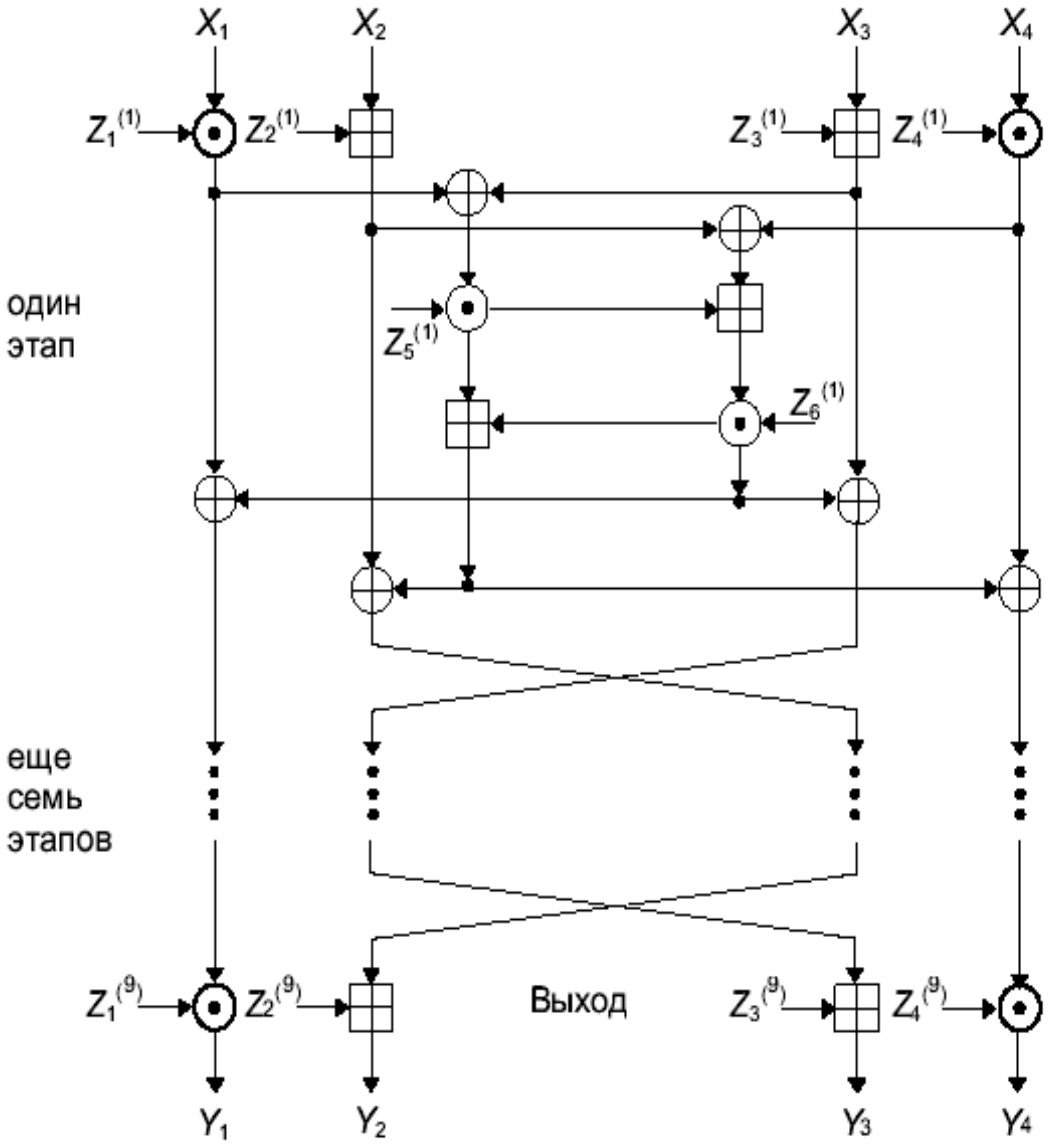
В приложениях не всегда просто заменить использование DES на IDEA, несмотря на надежность и скорость работы последнего. Например, если ваша база данных и шаблоны сообщений жестко запрограммированы на работу с 64-битовым ключом, реализации 128-битового ключа IDEA, вероятно, невозможна.

Для таких приложений можно создать 128-битовый ключ, объединив 64-битовый ключ с ним же самим. Однако, такая модификация заметно ослабляет алгоритм IDEA.

При условии, что скорость работы для вас более приоритетна, чем безопастность, можно попробовать вариант IDEA с меньшим числом раундов. В настоящее время лучший метод вскрытия IDEA работает быстрее любого вскрытия, только при использовании в алгоритме 2,5 и менее раундов. 4-раундовый алгоритм IDEA исполняется вдвое быстрее и, насколько известно, его надежность практически та же.

Использование IDEA

Стоит отметить, что в настоящее время IDEA широко используется в различных криптосистемах и приложениях. Однако его распространение осложнено главным образом тем, что IDEA запатентован в США и большинстве европейских стран, и его использование в коммерческих приложениях подразумевает лицензионные отчисления правообладателю. Этот факт в основном объясняет то, что IDEA пока не удалось вытеснить DES. Использование шифра IDEA в некоммерческих целях бесплатно.



X_i : 16-битовый подблок открытого текста
Y_i : 16-битовый подблок шифротекста
$Z_i^{(r)}$: 16-битовый подблок ключа
⊕ : побитовое "исключающее или" (XOR) 16-битовых подблоков
⊞ : сложение по модулю 2^{16} 16-битовых целых
⊙ : умножение по модулю $2^{16}+1$ 16-битовых целых при условии, что нулевой подблок соответствует 2^{16}

Литература

Б. Шнайер
Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – М.: Издательство ТРИУМФ, 2003

Ю. С. Харин, В. И. Берник, Г. В. Матвеев, С. В. Агиевич
Математические и компьютерные основы криптологии. – Мн.: Новое знание, 2003

Семенов Ю.А.
IDEA - международный алгоритм шифрования данных. – http://www.citforum.ru