3.6. Алгоритм IDEA

IDEA (International Data Encryption Algorithm) является блочным симметричным алгоритмом шифрования, разработанным Сюдзя Лай и Джеймсом Массей из швейцарского федерального института технологий. Первоначальная версия была опубликована в 1990 году. Пересмотренная версия алгоритма, усиленная средствами защиты от дифференциальных криптографических атак, была представлена в 1991году и подробно описана в 1992 году.

IDEA является блочным алгоритмом, который использует 128-битовый *ключ для*

Каждая операция *IDEA* выполняется над двумя 16-битными входами и создает один 16-битный выход. Этими операциями являются:

1. Побитовое исключающее OR, обозначаемое как $\oplus$.
2. Сумма целых по модулю 2^{16} (по модулю 65536), при этом входы и выходы трактуются как беззнаковые 16-битные целые. Эту операцию обозначим как +.
3. Умножение целых по модулю $2^{16} + 1$ (по модулю 65537), при этом входы и выходы трактуются как беззнаковые 16-битные целые, за исключением того, что блок из одних нулей трактуется как 2^{16}. Эту операцию обозначим как •.

Эти три операции являются несовместимыми в том смысле, что:

1. Не существует пары из трех операций, удовлетворяющих дистрибутивному закону. Например a • (b + c) <> (a • b) + (a • c)
2. Не существует пары из трех операций, удовлетворяющих ассоциативному закону. Например a + (b $\oplus$ c) <> (a + b) $\oplus$ c

Шифрование

Алгоритм IDEA состоит из восьми *раундов*, за которыми следует заключительное преобразование. Алгоритм разделяет блок на четыре 16-битных подблока. Каждый *раунд* получает на входе четыре 16-битных подблока и создает четыре 16-битных выходных подблока. Заключительное преобразование также получает на входе четыре 16-битных подблока и создает четыре 16-битных подблока. Каждый *раунд* использует шесть16-битных ключей, заключительное преобразование использует четыре *подключа*, т.е. всего в алгоритме используется 52 *подключа*.

Рисунок 3.7 – Алгоритм IDEA

Последовательность преобразований отдельного раунда

Рассмотрим последовательность преобразований отдельного *раунда*.

Одним из основных элементов алгоритма, обеспечивающих диффузию,является структура, называемая МА (умножение/сложение):

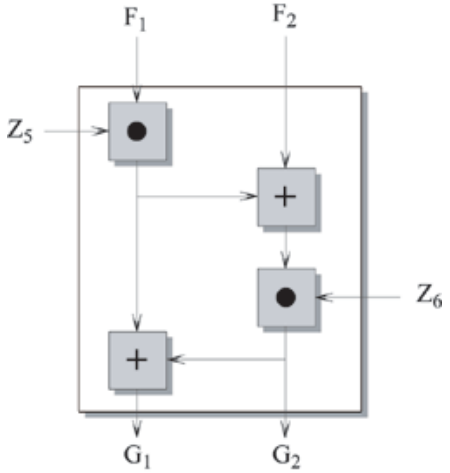

Рисунок 3.8 – Структура МА (умножение/сложение)

На вход этой структуре подаются два 16-битных значения и два 16-битных *подключа*, на выходе создаются два 16-битных значения.

Раунд начинается с преобразования, которое комбинирует четыре входных подблока с четырьмя *подключами*, используя операции сложения и умножения. Четыре выходных блока этого преобразования комбинируются,используя операцию XOR для формирования двух 16-битных

блоков,которые являются входами МА структуры. Кроме того, МА структура имеет на входе еще два *подключа* и создает два 16-битных выхода.

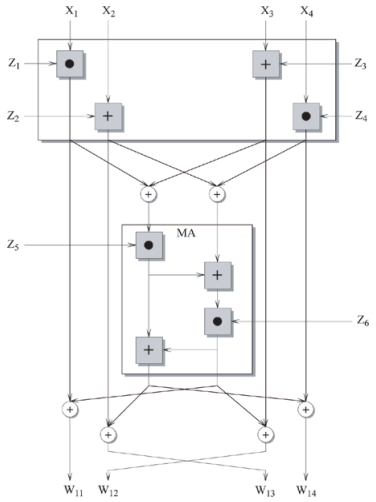

Рисунок 3.9 – I-ый раунд IDEA

В заключении четыре выходных подблока первого преобразования комбинируются с двумя выходными подблоками МА структуры, используя XOR для создания четырех выходных подблоков данной итерации. Заметим,что два выхода, которые частично создаются вторым и третьим входами (X_2 и X_3), меняются местами для создания второго и третьего выходов (W_{12} и W_{13}). Это увеличивает перемешивание бит и делает алгоритм более стойким для дифференциального криптоанализа.

Рассмотрим девятый *раунд алгоритма*, обозначенный как заключительное преобразование. Это та же структура, что была описана выше. Единственная разница состоит в том, что второй и третий входы меняются местами. Это сделано для того, чтобы дешифрование имело ту же структуру, что и шифрование. Заметим, что девятая стадия требует только четыре входных *подключа*, в то время как для первых восьми стадий для каждой из них необходимо шесть входных *подключей*.

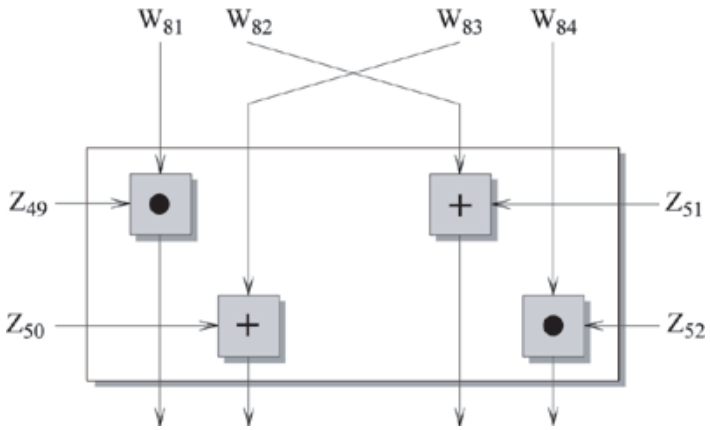

Рисунок 3.10 – Заключительное преобразование

Создание подключей

Пятьдесят два 16-битных *подключа* создаются из 128-битного *ключа шифрования* следующим образом. Первые восемь *подключей*, которые обозначим как Z_1, Z_2, ..., Z_8, получаются непосредственно из ключа, при этом Z_1 равен первым 16 битам, Z_2 равен следующим 16 битам и т.д. Затем происходит циклический сдвиг ключа влево на 25 бит, и создаются следующие восемь *подключей*. Эта процедура повторяется до тех пор, пока не будут созданы все 52 *подключа*.

Заметим, что каждый первый *подключ раунда* получен из своего подмножества бит ключа. Если весь ключ обозначить как $Z_{[1..128]}$, то первыми ключами в восьми *раундах* будут:

$$\mathbf{Z_1 = Z_{[1..16]}} \qquad \mathbf{Z_{25} = Z_{[76..91]}}$$
$$\mathbf{Z_7 = Z_{[97..112]}} \qquad \mathbf{Z_{31} = Z_{[44..59]}}$$
$$\mathbf{Z_{13} = Z_{[90..105]}} \qquad \mathbf{Z_{37} = Z_{[37..52]}}$$
$$\mathbf{Z_{19} = Z_{[83..98]}} \qquad \mathbf{Z_{43} = Z_{[30..45]}}$$

Хотя на каждом *раунде* за исключением первого и восьмого используются только 96 бит *подключа*, множество бит ключа на каждой итерации не пересекаются, и не существует отношения простого сдвига между *подключами* разных *раундов*. Это происходит потому, что на каждом *раунде* используется только шесть *подключей*, в то время как при каждой ротации ключа получается восемь *подключей*.

Дешифрование

Процесс дешифрования аналогичен процессу шифрования. Дешифрование состоит в использовании зашифрованного текста в качестве входа в ту же самую структуру *IDEA*, но с другим набором ключей. Дешифрующие ключи $U_1,...,U_{52}$ получаются из шифрующих ключей следующим образом:

1. Первые четыре *подключа* i-ого *раунда* дешифрования получаются из первых четырех *подключей* (10-i)-го *раунда* шифрования, где стадия заключительного преобразования считается 9-м *раундом*. Первый и четвертый ключи дешифрования эквивалентны мультипликативной инверсии по модулю $(2^{16} + 1)$ соответствующих первого и четвертого *подключей* шифрования. Для *раундов* со 2 по 8 второй и третий *подключи* дешифрования эквивалентны аддитивной инверсии по модулю (2^{16}) соответствующих третьего и второго *подключей* шифрования. Для *раундов* 1 и 9 второй и третий *подключи* дешифрования эквивалентны аддитивной инверсии по модулю (2^{16}) соответствующих второго и третьего *подключей* шифрования.
2. Для первых восьми *раундов* последние два *подключа* i *раунда* дешифрования эквивалентны последним двум *подключам* (9-i) *раунда* шифрования.

   Для мультипликативной инверсии используется нотация Z_j^{-1}, т.е.:

   $\mathbf{Z_j \bullet Z_j^{-1} = 1 \bmod (2^{16} + 1)}$

Так как $2^{16} + 1$ является простым числом, каждое ненулевое целое $Z_j <= 2^{16}$имеет уникальную мультипликативную инверсию по модулю ($2^{16} + 1$). Для аддитивной инверсии используется нотация ($-Z_j$), таким образом, мы имеем: $-Z_j + Z_j = 0 \bmod (2^{16})$

Для доказательства того, что алгоритм дешифрования с соответствующими *подключами* имеет корректный результат, рассмотрим одновременно процессы шифрования и дешифрования. Каждый из восьми *раундов* разбит на две стадии преобразования, первая из которых называется трансформацией, а вторая шифрованием.

При шифровании поддерживаются следующие соотношения на выходе трансформации:

$Y_1 = W_{81} \bullet Z_{49}$ $Y_3 = W_{82} + Z_{51}$

$Y_2 = W_{83} + Z_{50}$ $Y_4 = W_{84} \bullet Z_{52}$

Первая стадия первого *раунда* процесса дешифрования поддерживает следующие соотношения:

$J_{11} = Y_1 \bullet U_1$ $J_{13} = Y_3 + U_3$

$J_{12} = Y_2 + U_2$ $J_{14} = Y_4 \bullet U_4$

Подставляя соответствующие значения, получаем:

$J_{11} = Y_1 \bullet Z_{49}^{-1} = W_{81} \bullet Z_{49} \bullet Z_{49}^{-1} = W_{81}$

$J_{12} = Y_2 + -Z_{50} = W_{83} + Z_{50} = W_{83} + Z_{50} + -Z_{50} = W_{83}$

$J_{13} = Y_3 + -Z_{51} = W_{82} + Z_{51} + -Z_{51} = W_{82}$

$J_{14} = Y_4 \bullet Z_{52}^{-1} = W_{84} \bullet Z_{52} \bullet Z_{52}^{-1} = W_{84}$

Таким образом, выход первой стадии процесса дешифрования эквивалентен входу последней стадии процесса шифрования за исключением чередования второго и третьего блоков. Теперь рассмотрим следующие отношения:

$W_{81} = I_{81} \oplus MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{82} = I_{83} \oplus MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{83} = I_{82} \oplus MA_L(I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

$W_{84} = I_{84} \oplus MA_L(I_{81} \oplus I_{83}, I_{82} \oplus I_{84})$

Где $MA_R(X, Y)$есть правый выход MA структуры с входами X и Y, и $MA_L(X, Y)$ есть левый выход MA структуры с входами X и Y. Теперь получаем

$V_{11} = J_{11} \oplus MA_R (J_{11} \oplus J_{13}, J_{12} \oplus J_{14}) =$

$W_{81} \oplus MA_R(W_{81} \oplus W_{82}, W_{83} \oplus W_{84}) =$

$I_{81} \oplus MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}) \oplus$

$MA_R[I_{81} \oplus MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}) \oplus I_{83} \oplus$

$MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}), I_{82} \oplus$

$MA_L(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}) \oplus I_{84} \oplus$

$MA_L(I_{81} \oplus I_{83}, I_{82} \oplus I_{84})] =$

$I_{81} \oplus MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}) \oplus$

$MA_R(I_{81} \oplus I_{83}, I_{82} \oplus I_{84}) = I_{81}$

Аналогично мы имеем

$V_{12} = I_{83}$

$V_{13} = I_{82}$

$V_{14} = I_{84}$

Таким образом, выход второй стадии процесса дешифрования эквивалентен входу предпоследней стадии процесса шифрования за исключением чередования второго и третьего подблоков. Аналогично можно показать,что

$V_{81} = I_{11}$

$V_{82} = I_{13}$

$V_{83} = I_{12}$

$V_{84} = I_{14}$

Наконец, так как выход трансформации процесса дешифрования эквивалентен первой стадии процесса шифрования за исключением чередования второго и третьего подблоков, получается, что выход всего процесса шифрования эквивалентен входу процесса шифрования.