Weak Keys for IDEA

Joan Daemen, René Govaerts and Joos Vandewalle

Katholieke Universiteit Leuven, Laboratorium ESAT
Kardinaal Mercierlaan 94, B-3001 Heverlee, Belgium
email: joan.daemen@esat.kuleuven.ac.be

Abstract. Large classes of weak keys have been found for the block cipher algorithm IDEA, previously known as IPES [2]. IDEA has a 128-bit key and encrypts blocks of 64 bits. For a class of 2^{23} keys IDEA exhibits a linear factor. For a certain class of 2^{35} keys the cipher has a global characteristic with probability 1. For another class of 2^{51} keys only two encryptions and solving a set of 16 nonlinear boolean equations with 12 variables is sufficient to test if the used key belongs to this class. If it does, its particular value can be calculated efficiently. It is shown that the problem of weak keys can be eliminated by slightly modifying the key schedule of IDEA.

1 Introduction

At Eurocrypt '90 the block cipher proposal PES (Proposed Encryption Standard) was presented [1]. At Eurocrypt '91 the same authors presented a modification of PES, called IPES (Improved PES) [2]. The reason for this modification were new insights based on differential cryptanalysis [3]. IPES has become commercialized under the name IDEA (International Data Encryption Algorithm).

IDEA is an iterated cipher consisting of 8 similar rounds and a single output transformation. The building blocks of the round function are multiplication modulo $2^{16}+1$, addition modulo 2^{16} and bitwise XOR. IDEA has a 128-bit key and encrypts/decrypts data in blocks of 64 bits.

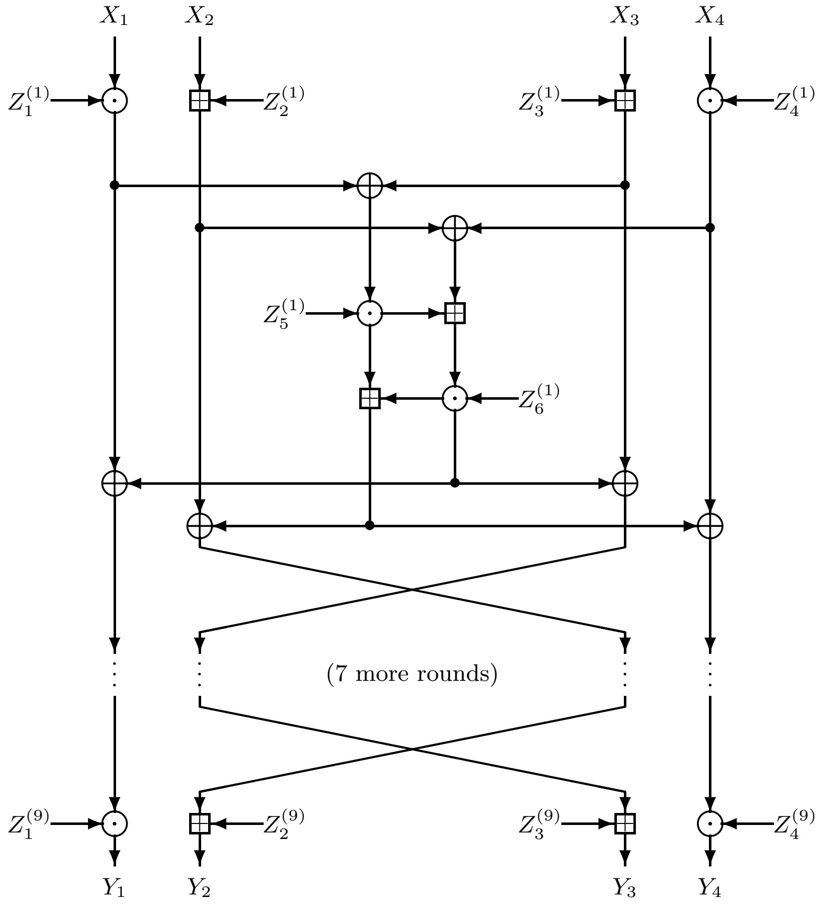
With exception of the key schedule, the IDEA decryption process is the same as its encryption process. The computational graph of the IDEA algorithm is shown in Fig.1. The encryption round keys are 16-bit substrings of the global key as specified in Table 1. The decryption round keys can be derived from the encryption round keys.

2 Linearities in the Modular Arithmetic Operations

Let x_i denote the i-th bit in the binary representation of the number X, i.e. $X = \sum 2^i x_i$. The bits of $Y = X + Z \bmod 2^n$ are given by

$$y_i = x_i \oplus z_i \oplus c_i \tag{1}$$

Appeared in *Advances in Cryptology – CRYPTO 1993*, Lecture Notes in Computer Science 773, D. R. Stinson (ed.), Springer-Verlag, pp. 224–231, 1993.



$X_i, Y_i, Z_i^{(r)}$: 16-bit plaintext, ciphertext and key subblocks
⊕ : bitwise XOR ⊞: addition mod 2^{16}
⊙ : multiplication mod $2^{16} + 1$ with 0000 (HEX) $\equiv 2^{16}$

Fig. 1. the encryption process of IDEA.

Appeared in *Advances in Cryptology – CRYPTO 1993*, Lecture Notes in Computer
Science 773, D. R. Stinson (ed.), Springer-Verlag, pp. 224–231, 1993.

r	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6
1	0–15	16–31	32–47	48–63	64–79	80–95
2	96–111	112–127	25–40	41–56	57–72	73–88
3	89–104	105–120	121–8	9–24	50–65	66–81
4	82–97	98–113	114–1	2–17	18–33	34–49
5	75–90	91–106	107–122	123–10	11–26	27–42
6	43–58	59–74	100–115	116–3	4–19	20–35
7	36–51	52–67	68–83	84–99	125–12	13–28
8	29–44	45–60	61–76	77–92	93–108	109–124
9	22–37	38–53	54–69	70–85	—	—

Table 1. Derivation of the encryption round keys of the global 128-bit key. The key bits are indexed starting from 0. The most significant bit (MSB) of the round keys are the bits with the lowest global index.

with c_i a carry bit that only depends on bits with indices *smaller than* i. The LSB of Y (y_0) is simply equal to $x_0 \oplus z_0$. Propagation of the MSBs of X and Z into Y is restricted to linear propagation (over $\mathbb{Z}_2^{16}$) into the MSB of Y.

For the multiplication by -1 (0000 HEX) modulo $2^{16}+1$ as defined in the IDEA block cipher it can easily be checked that

$$-1 \odot A = \bar{A} + 2 \bmod 2^{16} \tag{2}$$

with $\bar{A}$ the bitwise complement of A. Therefore multiplication by -1 inherits the linearity properties of the addition modulo 2^n.

3 Classes of Weak Keys yielding Linear Factors

The use of multiplicative subkeys with value 1 or -1 give rise to *linear factors* [4] in the round function. In the context of this paper a linear factor is a linear equation in key, input and output bits that holds for all possible *inputs*. The linear factors can be revealed by expressing the sum (modulo 2) of LSBs of the output subblocks of an IDEA round in terms of input and key bits.

As an example, we will express the XOR of the LSBs of the first and second output subblock of a round: $y_1 \oplus y_2$ (with the indices denoting the subblock number). From Fig.1 it can be seen that $y_1 \oplus y_2 = (X_1 \cdot Z_1)|_0 \oplus 1 \oplus x_3 \oplus z_3$. If $Z_1 = (-)1$, i.e. if the 15 MSB bits of the Z_1 are 0,

$$y_1 \oplus y_2 = x_1 \oplus x_3 \oplus z_1 \oplus z_3 \oplus 1 \ . \tag{3}$$

If the key bits are considered as (albeit unknown) constants, this linear factor can be interpreted as the propagation of knowledge from $x_1 \oplus x_3$ to $y_1 \oplus y_2$, denoted by $(1,0,1,0) \rightarrow (1,1,0,0)$. Similar factors and their corresponding conditions on subkey blocks can be found for all 15 combinations of LSB output bits and are listed in Table 2.

linear factor	Z_1	Z_4	Z_5	Z_6
$(0,0,0,1) \rightarrow (0,0,1,0)$	-	$(-)1$	-	$(-)1$
$(0,0,1,0) \rightarrow (1,0,1,1)$	-	-	$(-)1$	$(-)1$
$(0,0,1,1) \rightarrow (1,0,0,1)$	-	$(-)1$	$(-)1$	-
$(0,1,0,0) \rightarrow (0,0,0,1)$	-	-	-	$(-)1$
$(0,1,0,1) \rightarrow (0,0,1,1)$	-	$(-)1$	-	-
$(0,1,1,0) \rightarrow (1,0,1,0)$	-	-	$(-)1$	-
$(0,1,1,1) \rightarrow (1,0,0,0)$	-	$(-)1$	$(-)1$	$(-)1$
$(1,0,0,0) \rightarrow (0,1,1,1)$	$(-)1$	-	$(-)1$	$(-)1$
$(1,0,0,1) \rightarrow (0,1,0,1)$	$(-)1$	$(-)1$	$(-)1$	-
$(1,0,1,0) \rightarrow (1,1,0,0)$	$(-)1$	-	-	-
$(1,0,1,1) \rightarrow (1,1,1,0)$	$(-)1$	$(-)1$	-	$(-)1$
$(1,1,0,0) \rightarrow (0,1,1,0)$	$(-)1$	-	$(-)1$	-
$(1,1,0,1) \rightarrow (0,1,0,0)$	$(-)1$	$(-)1$	$(-)1$	$(-)1$
$(1,1,1,0) \rightarrow (1,1,0,1)$	$(-)1$	-	-	$(-)1$
$(1,1,1,1) \rightarrow (1,1,1,1)$	$(-)1$	$(-)1$	-	-

Table 2. Linear factors in the round function with conditions on the subkeys.

Multiple-round linear factors can be found by combining linear factors where the involved intermediate terms cancel out. For every round this gives conditions on subkeys that can be converted to conditions on global key bits using Table 1. An example is given in Table 3 for the global linear factor $(1,0,1,0) \rightarrow (0,1,1,0)$. The global key bits whose indices are given in this table must be 0. Since key bits with indices in 26-28, 72-74 or 111-127 don't appear, there are 2^{23} global keys that have this linear factor. This is called a *class of weak keys* since membership can easily be checked by observing some corresponding plaintext-ciphertext combinations.

round	input term	Z_1	Z_5
1	$(1,0,1,0)$	0–14	-
2	$(1,1,0,0)$	96–110	57–71
3	$(0,1,1,0)$	-	50–64
4	$(1,0,1,0)$	82–96	-
5	$(1,1,0,0)$	75–89	11–25
6	$(0,1,1,0)$	-	4–18
7	$(1,0,1,0)$	36–50	-
8	$(1,1,0,0)$	29–44	93–107
9	$(0,1,1,0)$	-	-

Table 3. Conditions on key bits for linear factor $(1,0,1,0) \rightarrow (0,1,1,0)$

Appeared in *Advances in Cryptology – CRYPTO 1993*, Lecture Notes in Computer
Science 773, D. R. Stinson (ed.), Springer-Verlag, pp. 224–231, 1993.

4 Classes of Weak Keys yielding Characteristics with Probability 1

In this section differential cryptanalysis [3] is applied where ‘difference’ is defined by bitwise XOR. The use of multiplicative subkeys with value 1 or -1 gives rise to characteristics with probability 1 in the round function.

A round is executed for a pair of inputs X and X^* with a given XOR $X' = X \oplus X^*$. Let ν be the 16-bit block 8000 (HEX), i.e. the MSB is 1 and all other bits are 0.

Suppose X and X^* only differ in the MSB bit of the 4-th subblock, hence $X'_1 = X'_2 = X'_3 = 0$ and $X'_4 = \nu$. If $Z_4 = (-)1$ this will still be the case after the application of Z_1 to Z_4. The left input to the MA structure is the same for X and X^*. The right input differs by ν. This XOR propagates unchanged through the top right (TR) addition to the bottom right (BR) multiplication by Z_6. If this is equal to $(-)1$, the output XOR is again ν. This difference propagates unchanged through the BL addition and the XORs to the 4 subblocks. The output difference Y' of the round is equal to $(\nu, \nu, \nu, 0)$. Hence if the 15 MSB of both Z_4 and Z_6 are 0, the input XOR $(0, 0, 0, \nu)$ gives rise to the output XOR $(\nu, \nu, \nu, 0)$ with probability 1, denoted by $(0, 0, 0, \nu) \Rightarrow (\nu, \nu, \nu, 0)$. A similar analysis can be made for any other of the 15 possible nonzero input XORs where only the MSB bits of the subblocks are allowed to be 1. The results are listed in Table 4.

characteristic	Z_1	Z_4	Z_5	Z_6
$(0,0,0,\nu) \Rightarrow (\nu,\nu,\nu,0)$	-	$(-)1$	-	$(-)1$
$(0,0,\nu,0) \Rightarrow (\nu,0,0,0)$	-	-	$(-)1$	$(-)1$
$(0,0,\nu,\nu) \Rightarrow (0,\nu,\nu,0)$	-	$(-)1$	$(-)1$	-
$(0,\nu,0,0) \Rightarrow (\nu,\nu,0,\nu)$	-	-	-	$(-)1$
$(0,\nu,0,\nu) \Rightarrow (0,0,\nu,\nu)$	-	$(-)1$	-	-
$(0,\nu,\nu,0) \Rightarrow (0,\nu,0,\nu)$	-	-	$(-)1$	-
$(0,\nu,\nu,\nu) \Rightarrow (\nu,0,\nu,\nu)$	-	$(-)1$	$(-)1$	$(-)1$
$(\nu,0,0,0) \Rightarrow (0,\nu,0,0)$	$(-)1$	-	$(-)1$	$(-)1$
$(\nu,0,0,\nu) \Rightarrow (\nu,0,\nu,0)$	$(-)1$	$(-)1$	$(-)1$	-
$(\nu,0,\nu,0) \Rightarrow (\nu,\nu,0,0)$	$(-)1$	-	-	-
$(\nu,0,\nu,\nu) \Rightarrow (0,0,\nu,0)$	$(-)1$	$(-)1$	-	$(-)1$
$(\nu,\nu,0,0) \Rightarrow (\nu,0,0,\nu)$	$(-)1$	-	$(-)1$	-
$(\nu,\nu,0,\nu) \Rightarrow (0,\nu,\nu,\nu)$	$(-)1$	$(-)1$	$(-)1$	$(-)1$
$(\nu,\nu,\nu,0) \Rightarrow (0,0,0,\nu)$	$(-)1$	-	-	$(-)1$
$(\nu,\nu,\nu,\nu) \Rightarrow (\nu,\nu,\nu,\nu)$	$(-)1$	$(-)1$	-	-

Table 4. XOR propagation in the round function with conditions on the subkeys.

The propagation of a given XOR for multiple rounds can be easily studied by letting the output XOR be the input XOR to the following round. The conditions on the subkeys can be read in Table 4.

An example for the plaintext XOR $(0, \nu, 0, \nu)$ is given in Table 5. It can be seen that for keys with only nonzero bits on positions 26–40, 72–76 and 108–122 the output XOR must be equal to $(0, \nu, \nu, 0)$. This is the largest class we found, comprising a total of 2^{35} keys. Membership can be checked by performing 2 encryptions where the plaintexts have a chosen difference and observing the difference in the ciphertexts. A similar table can be constructed for any input XOR consisting of ν and 0.

round	input xor	Z_4	Z_5
1	$(0, \nu, 0, \nu)$	48–62	-
2	$(0, 0, \nu, \nu)$	41–55	57–71
3	$(0, \nu, \nu, 0)$	-	50–64
4	$(0, \nu, 0, \nu)$	2–16	-
5	$(0, 0, \nu, \nu)$	123–9	11–25
6	$(0, \nu, \nu, 0)$	-	4–18
7	$(0, \nu, 0, \nu)$	84–98	-
8	$(0, 0, \nu, \nu)$	77–91	93–107
9	$(0, \nu, \nu, 0)$	-	-

Table 5. Propagation of plaintext XOR $(0, \nu, 0, \nu)$ in IDEA.

5 Expanding Classes of Weak Keys

Classes of weak keys can sometimes be significantly expanded at the cost of some more effort in the checking for membership. Omitting in Table 5 the conditions for the subkeys of round 8 gives rise to the class of 2^{51} keys with nonzero bits on positions 26–40, 72–83 and 99–122. We will show that both checking for membership and calculation of the specific key can be performed efficiently.

5.1 The Membership Test

The input XOR of round 8 is equal to the output XOR of round 7 and is guaranteed to be equal to $(0, 0, \nu, \nu)$ by the conditions on the subkeys of the first 7 rounds. Using the fact that $Z_3^{(9)}$, consisting of global key bits 54–69 is 0000 for these keys it can easily be derived that

$$Y_3' \oplus \nu = ({Z_1^{(9)}}^{-1} \cdot Y_1^*) \oplus ({Z_1^{(9)}}^{-1} \cdot Y_1) \ . \tag{4}$$

This can be verified by inspecting Fig.2. In (4) only $Z_1^{(9)}$ is unknown. This subkey consists of global key bits 22–37. For the given class only the 12 LSB may differ from 0. If the global key does not belong to the class of weak keys, the probability that (4) has a solution is 1/16. Additional encryptions can be performed to eliminate these solutions. Every pair of encryptions yields an equation for $Z_1^{(9)}$ similar to (4).

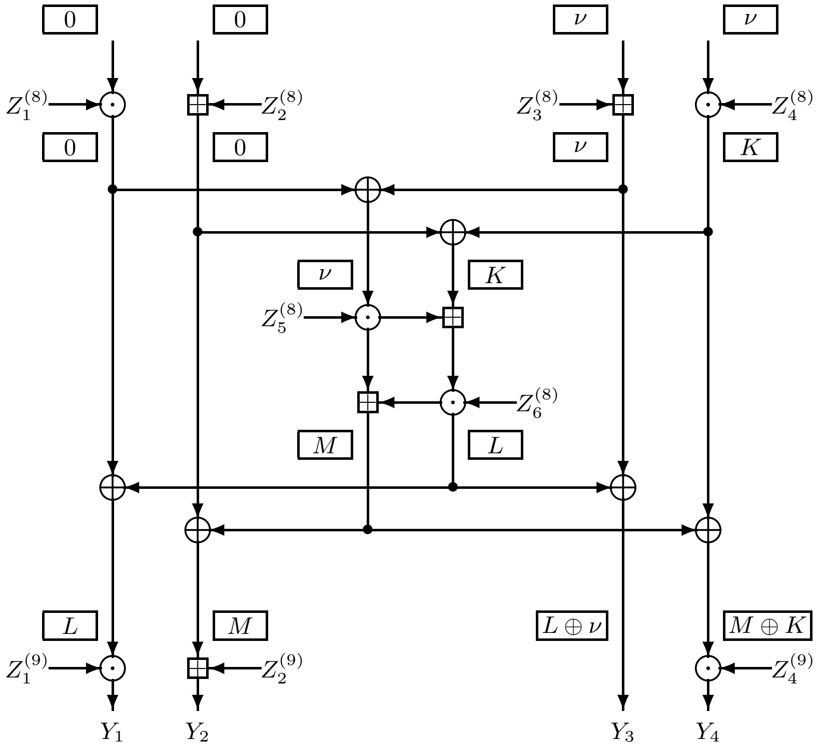


Fig. 2. XOR propagation of $X' = (0, \nu, 0, \nu)$ through the last round of IDEA for keys with only nonzero bits on positions 26–40, 72–83 and 99–122. The XORs are indicated in boxes.

Appeared in *Advances in Cryptology – CRYPTO 1993*, Lecture Notes in Computer Science 773, D. R. Stinson (ed.), Springer-Verlag, pp. 224–231, 1993.

5.2 The Determination of the Key

The value of the 12 unknown bits of $Z_1^{(9)}$ are already determined by the membership test. The following step is the determination of the 3 unknown bits of $Z_2^{(9)}$, the 12 unknown bits of $Z_4^{(9)}$ and the 7 unknown bits of $Z_4^{(8)}$. A consistency check can be executed on these bits in the following way. Suppose $Z_2^{(9)}$ and $Z_4^{(9)}$ are known. In this case it is possible to calculate the difference that is denoted by K in Fig.2. For this value K there must be a vector A (with MSB 0) such that

$$K = (Z_4^{(8)} \cdot A) \oplus (Z_4^{(8)} \cdot (A \oplus \nu)) = (Z_4^{(8)} \cdot A) \oplus ((Z_4^{(8)} \cdot A) + (Z_4^{(8)} \cdot 2^{15})) \quad (5)$$

For a given vector K it is easy to find the possible values of $Z_4^{(8)}$. Only values of $Z_4^{(8)}$ with the 9 LSB equal to 0 are valid. This information can be calculated in advance for every value of K and stored in an array of 2^{16} lists. The average number of possible $Z_4^{(8)}$ per K value turns out to be smaller than 1. Through this table, the observed value of K specifies a set of possible $Z_4^{(8)}$ values. If the set is empty, the chosen values for $Z_2^{(9)}$ and $Z_4^{(9)}$ must have been wrong. If the set is not empty, the K value resulting from another pair of encryptions (with input XOR at round 8 equal to $(0, 0, \nu, \nu)$) can be observed. The correct value for $Z_4^{(8)}$ must be in the list for both observed values of K. This can be repeated until there is no value for $Z_4^{(8)}$ left. The correct values for $Z_2^{(9)}$ and $Z_4^{(9)}$ are found if there is a value for $Z_4^{(8)}$ that is consistent for all the (say a maximum of 8) encryption pairs. Now 34 bits are fixed. The remaining 17 bits can easily be found by exhaustively trying all remaining 2^{17} possibilities and comparing it with any plaintext-ciphertext pair obtained during the attack.

The complete workload of the key determination is 16 chosen plaintext-difference encryptions, about 2^{15} modular additions, multiplications and table-lookups and 2^{17} key search encryptions.

6 A Modified IDEA Without Weak Keys

In the present specification of IDEA the conditions for weak multiplicative round keys are converted to the condition that global key bits must be 0. In Table 3 and 5 it can be seen that many global key bits appear more than once in the conditions.

Now let $\hat{Z}_i^{(r)} = \alpha \oplus Z_i^{(r)}$ with α a fixed nonzero binary vector. If in IDEA the subkeys $Z_i^{(r)}$ are replaced by $\hat{Z}_i^{(r)}$, the conditions for weak multiplicative keys are converted to the condition that some global key bits must be 0 and some must be 1. The vector α must be chosen such that for all potential multiple-round linear factors and characteristics, the conditions on the subkeys give conflicting conditions on global key bits. Because of the large overlap between subkeys, the exact value of α is not critical. For instance, for $\alpha =$ 0DAE (HEX) no weak keys were found.

7 Conclusions

Large classes of weak keys have been found for the block cipher IDEA. These keys are weak in the sense that it takes only a very small amount of effort to detect their use. It is possible to eliminate the weak key problem by slightly modifying the key schedule of IDEA.

References

[1] X. Lai and J.L. Massey, A Proposal for a New Block Encryption Standard, *Advances in Cryptology–Eurocrypt' 90*, Springer-Verlag, Berlin 1991, pp. 389–404.

[2] X. Lai, J.L. Massey and S. Murphy, Markov Ciphers and Differential Cryptanalysis, *Advances in Cryptology–Eurocrypt' 91*, Springer-Verlag, Berlin 1991, pp. 17–38.

[3] E. Biham and A. Shamir, Differential Cryptanalysis of DES-like Cryptosystems, *Journal of Cryptology*, Springer-Verlag, Vol. 4, No. 1, pp. 3–72, 1991.

[4] D. Chaum, J.-H. Evertse, Cryptanalysis of DES with a Reduced Number of Rounds, Sequences of Linear Factors in Block Ciphers, *Advances in Cryptology, Proceedings of Crypto 85*, pp. 192–211, 1985.

Appeared in *Advances in Cryptology – CRYPTO 1993*, Lecture Notes in Computer Science 773, D. R. Stinson (ed.), Springer-Verlag, pp. 224–231, 1993.