REPORTS IN INFORMATICS

ISSN 0333-3590

Trawling Twofish

Lars R. Knudsen

REPORT NO 189 **April 5, 2000**

Department of Informatics
UNIVERSITY OF BERGEN
Bergen, Norway

This report has URL http://www.ii.uib.no/publikasjoner/texrap/ps/2000-189.ps

Reports in Informatics from Department of Informatics, University of Bergen, Norway, is available at http://www.ii.uib.no/publikasjoner/texrap/.

Requests for paper copies of this report can be sent to:

Department of Informatics, University of Bergen, Høyteknologisenteret,
P.O. Box 7800, N-5020 Bergen, Norway

Trawling Twofish

Lars R. Knudsen

April 5, 2000

Abstract

Twofish is a 128-bit block cipher submitted as a candidate for the Advanced Encryption Standard (AES). It has a structure related to the Feistel structure and runs in 16 rounds. In this paper we consider mainly differentials of Twofish and show that there are differentials for Twofish for up to 16 rounds, predicting at least 32 bits of nontrivial information in every round. In addition, it holds that for any fixed user-selected key it is possible, at least in theory, to find one good pair of plaintexts following the differential through all 16 rounds. Also, we use these findings to distinguish Twofish reduced to 9 rounds (or fewer) from a randomly chosen permutation.

1 Introduction

Twofish [11] is a secret-key encryption primitive, which is one of the five final candidates for the Advanced Encryption Standard [10]. Twofish is a 16-round cipher which uses components from the ciphers Khufu [9], Square [1], and SAFER [8]. The 128-bit plaintexts are first split into four words of each 32 bits, $X^0_{LL}, X^0_{LR}, X^0_{RL}, X^0_{RR}$. The four words are then exclusive-or'ed with 32-bit round keys, K_0, K_1, K_2, K_3 respectively. Then we compute for $i = 0, \ldots, 15$:

$$\begin{aligned} w_1 &= g(X^i_{LL}) && (1)\\ w_2 &= g(X^i_{LR} << 8) && (2)\\ X^{i+1}_{LL} &= ((w_1 + w_2 + K_{2i+8}) \oplus X^i_{RL}) >> 1 && (3)\\ X^{i+1}_{LR} &= (w_1 + 2w_2 + K_{2i+9}) \oplus (X^i_{RR} << 1) && (4)\\ X^{i+1}_{RL} &= X^i_{LL} && (5)\\ X^{i+1}_{RR} &= X^i_{LR} && (6) \end{aligned}$$

The function $\mathbf{g}$ consists of four key-dependent S-boxes plus a linear transformation derived from an MDS-code. The key-dependent S-boxes are computed from two fixed 8-bit S-boxes q_0 and q_1. The ciphertext is the concatenation of the values $X^{16}_{RL} \oplus K_4, X^{16}_{RR} \oplus K_5, X^{16}_{LL} \oplus K_6, X^{16}_{LR} \oplus K_7$. Figure 1 is a non-detailed picture of one round of Twofish. Here $\mathbf{g}_8$ is the same as $\mathbf{g}$ but where the inputs are rotated by eight positions to the left. For a more detailed pictorial illustration of the encryption function of Twofish we refer to Figure 1 of [11].

It follows by inspection of Figure 1 that Twofish does not have the classical Feistel structure as claimed. If one removes the one-bit rotations, then the resulting cipher is a Feistel cipher. Although it is possible to incorporate the one-bit rotations inside the round function by applying simple transformations, this would result in different round functions in the different rounds.

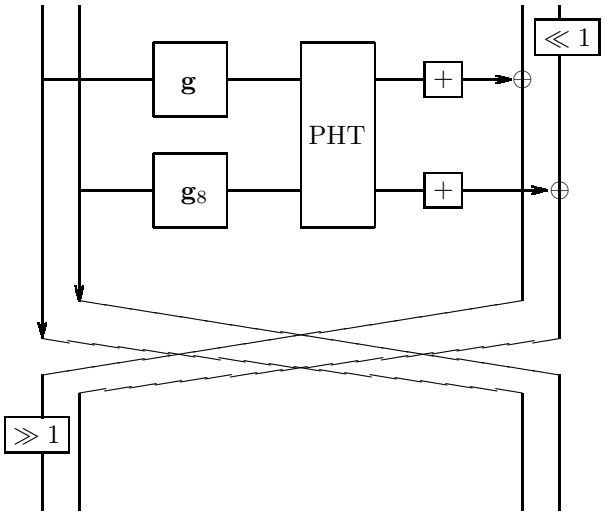

Figure 1: The Twofish graph. $\boxed{+}$ denotes the addition of a round key modulo 2^{32}, $\mathbf{g}_8$ is the function $\mathbf{g}$ but where the inputs are rotated 8 positions to the left, and $\mathrm{PHT}(x, y) = (x + y, x + 2y)$ both outputs modulo 2^{32}.

2 The Twofish S-boxes q_0 and q_1

In this section we analyse the fixed S-boxes used in Twofish. Each of the S-boxes q_0 and q_1 are constructed from four 4-bit S-boxes, t_1, t_2, t_3, and t_4. Call this construction the $q_{0,1}$-construction. The following fact is well-known.

Fact 1 *Consider a function $f : \{0,1\}^r \to \{0,1\}^r$. If f is a permutation, then the algebraic degree of any output bit as a function of the input bits is at most $r - 1$.*

Now we can prove the following property of the S-boxes.

Fact 2 *For any choices of the bijective 4-bit S-boxes t_1, t_2, t_3, and t_4 in the $q_{0,1}$-construction each bit in the output of the resulting 8-bit S-box can be written as a function of the input bits with algebraic degree at most six.*

Proof. The left half of the 8-bit input in the $q_{0,1}$-construction maps one-to-one to both the left and right halves of the output, and similarly for the right half of the input. Therefore, any output bit will be of at most degree three as a function of either half of the input, totally at most degree six. □

Note that the nonlinear order of an S-box is not the same as the algebraic degree of the output bits. The nonlinear order of an n-bit bijective S-box is the minimum algebraic degree of the $2^n - 1$ boolean functions obtained from a linear combination of the n coordinate functions. It can be shown that the nonlinear order of a randomly chosen bijective n-bit S-box is $n - 2$ with a high probability.

The authors of Twofish note [11, §7.2.1] "The construction method for building q_0 and q_1 from 4-bit permutations was chosen because .... without adding any apparent weaknesses to the cipher". For a randomly chosen 8-bit permutation the probability is very high that the algebraic degree of one or more output bits as functions of the input bits is seven. So random 8-bit S-boxes have better properties than the

$q_{0,1}$-construction with respect to the algebraic degrees, the question is if they are substantially better.

3 The linear transformation

The linear transformation inside the function **g** of Twofish is similar to the constructions in Square [1] and Rijndael [2]. It is a permutation of a vector of four bytes, such that for any two different input vectors, the total number of different bytes in the input vector and the output vector is at least five. This provides diffusion to the cipher. Two inputs to **g** different in only one of the four bytes are guaranteed to differ in all four bytes at the output of **g**. However, this also enables an attacker to specify a so-called "truncated differential", see e.g. [3, 4, 7], of probability one through the **g** transformation. Let (x, y, z, w) denote the difference of two vectors each of four bytes for any definition of difference, and let $(x, y, z, w) \xrightarrow{g} (X, Y, Z, W)$ denote that two input vectors of differences x, y, z, and w in the four bytes can lead to outputs of differences X, Y, Z, and W in the four bytes. Then the diffusion property implies that for $a \neq 0$ the differential $(a, 0, 0, 0) \xrightarrow{g} (b_0, b_1, b_2, b_3)$ where $b_i \neq 0$ for $i = 0, \ldots 3$ holds with probability one.

4 Differentials for Twofish

In this section we consider (truncated) differentials of Twofish. The differentials will specify the expected differences in each of the 32-bit words in the intermediate ciphertexts. Let us introduce some notation. We define the difference between two 32-bit words, X and Y as

$$X - Y \bmod 2^{32}.$$

With this definition, the difference before and after the addition of a round key is the same. Also, the PHT-transform is linear with respect to this difference.

We shall write a one-round differential as

$$(a, b, c, d) \rightarrow (e, f, a, b) : (a, b) \rightarrow (i, j) \rightarrow (k, l), \tag{7}$$

where all small letters denote a difference of two 32-bit words. The first two tuples represent the differences in the four input words and in the four output words of the particular round, see Figure 2. Here $k = i + j \bmod 2^{32}$ and $l = k + j \bmod 2^{32}$. The three following pairs specify first the differences in the inputs to the **g**-functions, then the differences in the words before and after the PHT-transform, respectively. Also, we shall only be interested in whether the difference in a 32-bit word is zero or nonzero.

Here we give a two-round differential, Ω_1, which has probability 2^{-32} and gives non-trivial information about at least 64 bits of the (intermediate) ciphertexts.

$$\begin{array}{lcll} (a, 0, c, d) \rightarrow (e, f, a, 0) & : & (a, 0) \rightarrow (i, 0) \rightarrow (i, i), & p = 1 \\ (e, f, a, 0) \rightarrow (h, 0, e, f) & : & (e, f) \rightarrow (2m, -m) \rightarrow (m, 0), & p = 2^{-32} \end{array}$$

The differential is iterative, that is, it can be concatenated with itself any number of times. If we start and end with one-round differentials of probability one, this yields $2r+1$-round differentials of probability $2^{-(32r)}$. Some explanation of the differential. The only requirement on the input differences is that $(a, c, d) \neq (0, 0, 0)$. Then $a \neq 0 \Rightarrow i \neq 0$, and $e \neq 0, f \neq 0 \Rightarrow m \neq 0$. In the first round, two inputs of nonzero

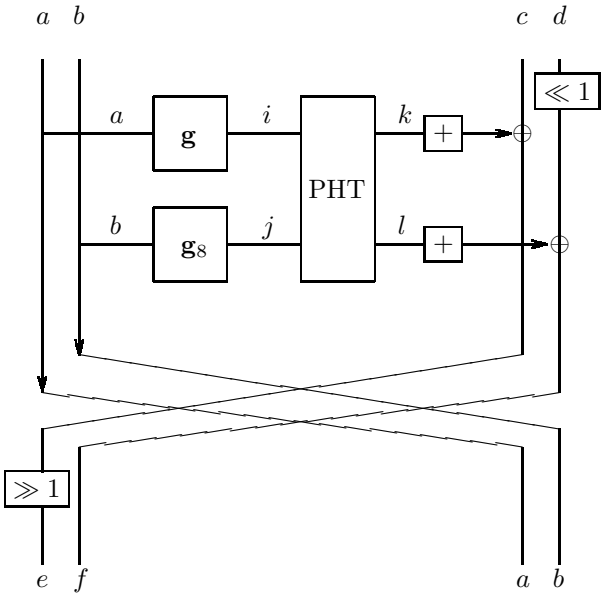

Figure 2: A one-round truncated differential.

difference a to $\mathbf{g}$ yields some nonzero difference i in the outputs of $\mathbf{g}$, and since the PHT is linear with respect to the difference used, it follows that the differences in both 32-bits words after addition of the round keys will be i. Note that since the outputs of the round function are combined with the right halves of the inputs to the round using the exclusive-or operation, the differences e and f are not just $c \oplus i$ and $d \oplus i$, respectively. More precisely, let e_1 and e_2 be the two texts of difference e and similarly for the other words. Then $e = e_1 - e_2 = (i_1 \oplus c_1) - (i_2 \oplus c_2)$ (here we ignored the one-bit rotations), where $i = i_1 - i_2$ and $c = c_1 - c_2$. However, if the values of c and d are nonrandom and related, then so are the values of e and f. This phenonmenon will be discussed later in this paper. In the second round, the two pairs of inputs of differences e and f, respectively, lead to differences $2m$ and $-m$ with an average probability of 2^{-32}, where we assumed that neither e nor f is zero, which will happen with probability $(1 - 2^{-31})$ if c, d are random. Note also, that $(e, f) = (0, 0)$ with probabilty 2^{-64} if c, d are random, in which case the (rest of the) second round has a probability of one. All in all, the probability of the second round is approximately 2^{-32}.

Summing up, the differential predicts 32 bits of information in each of two rounds, a total of 64 bits with a probability of (about) 2^{-32}.

For Ω_1 a pair of plaintexts will have a difference of $(a, 0, c, d)$, where $(a, c, d) \neq (0, 0, 0)$. Thus, it is possible to generate

$$\left(\begin{array}{c} 2^{96} \\ 2 \end{array} \right) 2^{32} \approx 2^{223}$$

pairs of plaintexts of this difference. A 15-round differential will have a probability of 2^{-224}.

Also, there is the following 2-round iterative differential, Ω_2 of probability 2^{-32}

$$\begin{array}{lcll} (0, b, c, d) \rightarrow (e, f, 0, b) & : & (0, b) \rightarrow (0, i) \rightarrow (i, 2i), & p = 1 \\ (e, f, 0, b) \rightarrow (0, h, e, f) & : & (e, f) \rightarrow (-m, m) \rightarrow (0, m), & p = 2^{-32}. \end{array}$$

Iterated to 15 rounds also Ω_2 will have a probability of 2^{-224}. There are 2^{223} pairs of plaintexts with the desired difference. Therefore there are totally 2^{224} pairs of

plaintexts for Ω_1 and Ω_2. Altogether, for any fixed key, one can expect at least one good pair for one of the differentials, Ω_1, Ω_2 for Twofish up to 15 rounds, that is, at least one pair of plaintexts which follows the expected values in the differential in each round. The first four rows of Table 1 give the probabilities of the differentials, the number of plaintexts needed to generate one good pair, and the total number of good pairs for Twofish reduced to 9, 11, 13, and 15 rounds.

The above differentials can be extended by using a 1-round differential of probability one in the first round. One can use the following differential, $\Omega_{1,0}$

$$(0,0,a,0) \to (b,0,0,0) \quad : \quad (0,0) \to (0,0) \to (0,0), \quad p = 1$$

which can be concatenated with Ω_1. And the following differential, $\Omega_{2,0}$

$$(0,0,0,b) \to (0,c,0,0) \quad : \quad (0,0) \to (0,0) \to (0,0), \quad p = 1$$

can be concatenated with Ω_2. This yields $(2r+2)$-round differentials of probabilities $2^{-(32r)}$. The "price" to pay for this improvement in probability (or one extra round) is fewer pairs of plaintexts with desired difference. There are $2^{63}2^{96} = 2^{159}$ pairs of plaintexts for each of the two differentials with the desired input differences, a total of 2^{160} pairs. Thus for any fixed key, one can expect to get at least one good pair for Twofish reduced to 12 or fewer rounds. The middle three rows of Table 1 give the probability of the differentials, the number of plaintexts to generate one good pair, and the total number of expected good pairs. In an attempt to get access to more

# Rounds	Probability	# Plaintexts to generate 1 good pair	Total no. good pairs	Differentials
9	2^{-128}	2^{64}	2^{96}	Ω_1^* and Ω_2^*
11	2^{-160}	2^{80}	2^{64}	
13	2^{-192}	2^{96}	2^{32}	
15	2^{-224}	2^{128}	1	
8	2^{-96}	2^{64}	2^{64}	$\Omega_{1,0} \mid \Omega_1^*$, and $\Omega_{2,0} \mid \Omega_2^*$
10	2^{-128}	2^{96}	2^{32}	
12	2^{-160}	2^{128}	1	
8	2^{-128}	2^{64}	2^{128}	$\Omega_{1,1} \mid \Omega_1^*$, and $\Omega_{2,1} \mid \Omega_2^*$
10	2^{-160}	2^{80}	2^{96}	
12	2^{-192}	2^{96}	2^{64}	
14	2^{-224}	2^{112}	2^{32}	
16	2^{-256}	2^{128}	1	

Table 1: Truncated differentials for different number of rounds of Twofish. The second column is the probability for each one of two differentials, the third column the number of plaintexts required to generate one good pair, and the fourth column the expected total number of good pairs. Ω^* means Ω concatenated with itself some number of times, '|' means concatenation of differentials.

pairs of plaintexts to be used in the analysis we introduce some more differentials. The following differential, $\Omega_{1,1}$

$$(u,v,w,x) \to (a,0,u,v) \quad : \quad (u,v) \to (u',v') \to (y,z), \quad p = 2^{-32}$$

can be concatenated with Ω_1 and the following differential, $\Omega_{2,1}$

$$(u,v,w,x) \to (0,b,u,v) \quad : \quad (u,v) \to (u',v') \to (y,z), \quad p = 2^{-32}$$

can be concatenated with Ω_2. In both cases one gets $2r$-round differentials of probability 2^{-32r}. The advantage of this approach is that more pairs of plaintexts can be used. There are approximately 2^{255} pairs of plaintexts with the desired difference.

The last five rows of Table 1 give the probabilities, the number of plaintexts to get one good pair, and the expected number of total good pairs. As seen, for Twofish with the full 16 rounds, for any fixed key, one can expect to get one right pair following one of the two differentials. This does not necessarily mean that such good pairs can be exploited in a cryptanalytic attack. However it is surprising, in our opinion, that it is possible to push non-trivial information through all 16 rounds of Twofish.

5 Distinguishing Twofish from a random permutation

In this section we use the results of the previous section to distinguish Twofish reduced to 9 or fewer rounds from a randomly chosen permutation.

Consider Table 1 and the entry for 9 rounds. Each of the differentials has a probability of 2^{-128} and there are 2^{223} possible pairs of plaintexts for each of them satisfying the input difference. For any fixed key of Twofish reduced to 9 rounds, for a pair of plaintexts with the input difference, one can expect a pair of ciphertexts to have equal values in one (particular) 32-bit word with a probability of $2^{-32}+2^{-128}$. With 2^{223} pairs of plaintexts for each of the differentials, one can expect to find totally $2^{192} + 2^{96}$ good pairs, while for a randomly chosen permutation one finds "only" 2^{192} good pairs, in both cases with a standard deviation of 2^{96}. In a distinguishing attack, one would count the number of good pairs, and say "random" if the number is less than $2^{192} + 2^{95}$ and "Twofish - 9 rounds" otherwise. Then it can be shown that the probability of successfully distinguishing Twofish from random is about 70%. For this distinguisher we need all 2^{128} (known!) plaintexts.

For Twofish with fewer than nine rounds, the probabilities of the differentials will be larger and consequently distinguishing attacks based on these differentials will have a lower complexity.

The distinguishing attacks can be extended to include key-recovery. The complexities of such attacks are left open.

6 Open problems and future work

Consider the differential Ω_1, restated here for convenience.

$$\begin{array}{lcll} (a,0,c,d) \to (e,f,a,0) & : & (a,0) \to (i,0) \to (i,i), & p = 1 \\ (e,f,a,0) \to (h,0,e,f) & : & (e,f) \to (2m,-m) \to (m,0), & p = 2^{-32} \end{array}$$

As mentioned above the values of e and f cannot be determined directly from the values of c, d, and i. This is because the differences considered here are defined by subtraction modulo 2^{32}, but plaintext halves are combined with the exclusive-or operation. (In addition there is a one-bit rotation affecting the value of e.)

However, the values of e and f are not random for given c, d, and i. To illustrate this, let us consider a modified variant of Twofish. Instead of combining the plaintext

halves via the exclusive-or operation assume that the halves are added (word-wise) modulo 2^{32}. Also, we shall ignore the one-bit rotations. Note that this yields a valid block cipher. In this case the 2-round differential will be

$$\begin{array}{lcll} (a, 0, c, d) & \rightarrow & (c+i, d+i, a, 0) & p = 1 \\ (c+i, d+i, a, 0) & \rightarrow & (a+m, 0, c+i, d+i) & p = 2^{-32}, \end{array}$$

where we have omitted to specify the differences inside the round function. In this case the differential predicts not only 32 bits of information in the round function of each round, but totally 64 bits in the ciphertexts. Thus, for such a variant of Twofish, distinguishing attacks similar to those discussed in the previous section will have an increased performance. Note that $c - d$ can be assumed to be known in a known or chosen plaintext attack. This property iterates to any number of rounds. It is well-known that the exclusive-or operation and addition modulo 2^{32} are closely related with respect to differentials, see e.g. [6, 5], therefore the values of e and f will depend on the values of c, d and i. It seems our findings in this paper can only be improved by incorporating such dependencies. Such an analysis is problematic due the mixed use of group operations.

In the differential analysis and in the distinguishing attacks we have not taken advantage of any intrinsic properties of $\mathbf{g}$ and $\mathbf{g}_8$. In the above analysis we assumed that these are randomly chosen 32-bit permutations. However, as shown in Sections 2 and 3 there are properties of the Twofish round permutation which are not present in a randomly chosen permutation. We are convinced that the attacks as described above work despite of these facts, and also that incorporating intrinsic properties of $\mathbf{g}$ and $\mathbf{g}_8$ will only improve on our results, not the opposite. Finally, we note that the designers themselves[11, §8.3.1] have found some interesting high-probability truncated differentials through the whole round function, called F in [11].

7 Conclusion

In this paper we analysed the AES-candidate Twofish. We showed that it is possible to distinguish Twofish reduced to 9 (or fewer) rounds from a randomly chosen permutation. Thus, the level of security seems to be lower than anticipated by the designers in [12] where it is claimed that a maximum of six rounds can be cryptanalysed.

Also, we have shown that there exist differentials for Twofish for up to 16 rounds, predicting at least 32 bits of nontrivial information in every round. Moreover, the probabilities of these differentials are high enough such that one can expect to find one good pair of plaintexts following the differential through all 16 rounds for any fixed key. This is mainly due to the structure of the round function of Twofish. It is possible to separate the two halves of the texts inside the round function with non-trivial probability. Our analysis has not made use of any intrinsic properties of the S-boxes and linear transformations in Twofish. We believe that our findings will only be improved taking such an approach.

Acknowledgments

The author would like to thank Don Coppersmith, Willi Meier, Håvard Raddum, and Vincent Rijmen for helpful comments.

References

[1] J. Daemen, L. Knudsen, and V. Rijmen. The block cipher Square. In E. Biham, editor, *Fast Software Encryption, Fourth International Workshop, Haifa, Israel, January 1997, LNCS 1267*, pages 149–165. Springer Verlag, 1997.

[2] J. Daemen and V .Rijmen. AES proposal: Rijndael. Submitted as an AES Candidate Algorithm. Description available from NIST, see `http://www.nist.gov/aes`.

[3] L.R. Knudsen. New potentially weak keys for DES and LOKI. In A. .De Santis, editor, *Advances in Cryptology: EUROCRYPT'94, LNCS 950*, pages 419–424. Springer Verlag, 1995.

[4] L.R. Knudsen and T. Berson. Truncated differentials of SAFER. In Gollmann D., editor, *Fast Software Encryption, Third International Workshop, Cambridge, UK, February 1996, LNCS 1039*, pages 15–26. Springer Verlag, 1995.

[5] L.R. Knudsen and W. Meier. Improved differential attack on RC5. In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO'96, LNCS 1109*, pages 216–228. Springer Verlag, 1996.

[6] L.R. Knudsen, V. Rijmen, R.L. Rivest, and M.P.J. Robshaw. On the design and security of RC2. In S. Vaudenay, editor, *Fast Software Encryption, Fift International Workshop, FSE'98, Paris, France, March 1998, LNCS 1372*, pages 206–221. Springer Verlag, 1998.

[7] L.R. Knudsen, M.P.J. Robshaw, and D. Wagner. Truncated differentials and Skipjack. In M. Wiener, editor, *Advances in Cryptology: CRYPTO'99, LNCS 1666*, pages 165–180. Springer Verlag, 1999.

[8] J.L. Massey. SAFER K-64: A byte-oriented block-ciphering algorithm. In R. Anderson, editor, *Fast Software Encryption - Proc. Cambridge Security Workshop, Cambridge, U.K., LNCS 809*, pages 1–17. Springer Verlag, 1994.

[9] R. Merkle. Fast software encryption functions. In A.J. Menezes and S.A. Vanstone, editors, *Advances in Cryptology - CRYPTO'90, LNCS 537*, pages 476–501. Springer Verlag, 1991.

[10] National Institute of Standards and Technology. Advanced encryption algorithm (AES) development effort. `http://www.nist.gov/aes`.

[11] Schneier, Kelsey, Whiting, Wagner, Hall, and Ferguson. Twofish: A 128-bit block cipher. Submitted as candidate for AES Available at `http://www.counterpane.com/twofish.html`.

[12] Schneier, Kelsey, Whiting, Wagner, and Ferguson. Comments on Twofish as an AES candidate. Document, March 24, 2000. To be presented at AES3, April 2000.