Twofish Technical Report #1

Upper bounds on differential characteristics in Twofish

Niels Ferguson
Counterpane Systems
niels@counterpane.com

August 17, 1998

Abstract

In [SK+98] the Twofish block cipher was introduced, and initial estimates of an upper bounds on the probability of a 12-round differential were given. These results used an imperfect model of Twofish. We present an improved model, and show that any 12-round differential characteristic has a probability of at most $2^{-102.8}$.

Keywords: Twofish, cryptography, cryptanalysis, block cipher, AES.
Current web site: http://www.counterpane.com/twofish.html

1 Introduction

Twofish was introduced in [SK+98]. That report contained an initial analysis of the feasibility of a differential attack against the Twofish cipher. In this paper we will investigate differential attacks against Twofish further. We assume familiarity with both Twofish and differential cryptanalysis.

The results of [SK+98] are not hard as the model used to estimate the best differential is only an approximation of Twofish. We started a project to investigate differential attacks against Twofish further. This paper is a status report of our results to date. We expect to continue this work and achieve significant improvements over our current results.

The first choice we have to make in differential cryptanalysis is what type of differences to use. Twofish contains S-boxes, an MDS matrix multiply, addition modulo 2^{32}, XORs, and rotations. There are two types of differences that we think could be useful: a XOR difference, and a difference mod 2^{32}. When we use a XOR difference we have to use approximations for the S-boxes and the additions modulo 2^{32}; when we use a differences modulo 2^{32} we have to use approximations for the S-boxes, the MDS matrix multiply, the XORs, and the rotations.

The XOR and addition operations are fairly closely related, and either operation can be approximated with reasonable success in the group of the other operation. In the comparison we will ignore the S-boxes; we assume they are equally hard to approximate in each of the groups. A XOR differential has to approximate 2 addition operations in each round. A additive differential has to approximate the MDS matrix, a single XOR, and a rotation. We estimate that it is about as difficult to approximate an addition for a XOR differential as it is to approximate a XOR for an additive differential. The rotation seems to be somewhat easier to approximate for an additive differential than a XOR operation. So if we ignore the MDS matrix, it would seem that additive differentials are more attractive.

For our analysis, the MDS matrix multiply is best written as a linear function: each output bit is the

XOR of several input bits. This is very easy from the point of view of a XOR difference; no approximation is necessary and any given input difference leads to precisely one output difference. For an additive difference this is much harder. There do not seem to be any good approximations of the MDS matrix for an additive difference. Therefore, we estimate that XOR based differentials are much more effective than additive differentials. In the rest of this paper we will only look at XOR based differentials.

1.1 Notation

We use the definitions and symbols of the Twofish report [SK+98]. Let $\mathcal{B}$ be the set of all possible byte values. Let G be the F-function without the key-dependent S-boxes. Thus G consists of two MDS matrix multiplies, the PHT, and the subkey addition.

2 Differentials of the S-boxes

In this section we look at differential characteristics of the S-boxes. Each S-box consists of a sequence of q-mappings and XORs with a key byte. For q_0 and q_1 the probability of each differential can easily be computed by trying all possible pairs of inputs.

We define $p_i(a, b)$ to be the probability that q_i has an output difference of b, given an input difference of a. In other words:

$$p_i(a, b) := \Pr_{x \in \mathcal{B}} [q_i(x \oplus a) = q_i(x) \oplus b] \qquad i = 0, 1$$

We now look at the first two stages of an S-box. This consists of a q-mapping, followed by a XOR with a key byte, followed by another q-mapping. As usual, we assume uniform random distributions of the input values and the key bytes. We define $p_{ij}(a, b)$ to be the probability that this construction gives an output difference b given an input difference a, where i is the number of the first q-mapping, and j is the number of the second q-mapping. It is easy to see that

$$p_{ij}(a, b) = \sum_{d \in B} p_i(a, d) p_j(d, b) \qquad (1)$$

for $i, j \in \{0, 1\}$, and we can extend this definition to arbitrary long chains of q-mappings and key-byte XORs. In general it holds that

$$p_{ij\ldots m}(a, b) = \sum_{d \in B} p_i(a, d) p_{j\ldots m}(d, b)$$

for $i, j, \ldots, m \in \{0, 1\}$. This allows us to compute the exact probabilities for each of the S-boxes in Twofish. Table 1 gives the probabilities of the best differential of each of the S-boxes for each of the key length. From this point of view the S-boxes are very good; there are no high-probability differentials. (Note that the average differential probability is $1/255 = 1.0039 \cdot 2^{-8}$ as we know that the S-boxes are permutations and thus the output differential 0 does not occur in non-trivial cases. The best differential probability must be at least as large as the average.)

Note that the numbers in this table hold only when the key bytes are chosen at random. If we try a differential many times, each time with random input and key byte values then we expect to get the numbers in the table. However, for any particular set of key bytes there are differentials with a much higher probability (as shown in [SK+98]). Our computations are no longer valid because for any fixed key byte, the differential probabilities of p_i and p_j in equation 1 are no longer independent of each other.

Twofish uses the same S-boxes in each round. When analysing a multi-round differential characteristic the differential probabilities of each of the round functions are not independent either. This makes the analysis of the probability of a differential characteristic more difficult.

3 Differentials of F

The function F takes a 64-bit input and produces a 64-bit output. Thus there are a total of about 2^{128} possible differentials. It is clearly not possible to compute or list all of them. To alleviate this problem we will group the differentials in sets, and for every set compute upper bounds on the probability of the differentials in that set.

We split the 2^{128} different differential patterns into a number of subsets. The input difference is classified by the set of input-bytes that are non-zero. There are 256 different classifications of input differences. The output difference too are classified by the set of output-bytes that are non-zero. We group differentials with the same input and output classification in the same set. There are therefore 2^{16} different sets of differentials, each containing between 1 and 255^{16} elements.

We will construct differentials of F in two steps. First we use a differential approximation of the S-boxes, and then we use an approximation of the differentials of G.

	128-bit key	192-bit key	256-bit key
Sbox 0	$1.0649 \cdot 2^{-8}$	$1.0084 \cdot 2^{-8}$	$1.0043 \cdot 2^{-8}$
Sbox 1	$1.0566 \cdot 2^{-8}$	$1.0087 \cdot 2^{-8}$	$1.0043 \cdot 2^{-8}$
Sbox 2	$1.0533 \cdot 2^{-8}$	$1.0097 \cdot 2^{-8}$	$1.0045 \cdot 2^{-8}$
Sbox 3	$1.0538 \cdot 2^{-8}$	$1.0088 \cdot 2^{-8}$	$1.0044 \cdot 2^{-8}$

Table 1: Best differential probabilities of the S-boxes

3.1 Differentials of G

The MDS matrix multiply is purely linear, and thus creates no problem for our differential. The PHT and key addition use addition modulo 2^{32} as basic operation. This makes the differentials non-trivial. A theoretical analysis of differential probabilities is difficult as the probabilities at the result are not independent of each other. We therefore chose to use numerical simulation to establish bounds on the differential probability.

We are trying to derive an upper bound on differential probabilities. Therefore, we are interested in finding good bounds for the most likely differentials of G. In [SK+98] it is shown that for any 128-bit key, the best differential probability of an S-box is 18/256. If we look only at the S-boxes, then the most likely differentials occur when there are a low number of active S-boxes. The most important task is thus to find good bounds on the differential characteristics of G for differentials with a low number of active S-boxes.

We performed numerical simulations of differentials of G. Given an input difference, we generated n random input pairs with that difference and applied the G function (using random keys). We collected the output differences and counted how often each of them appeared. Due to limited resources we could only do this analysis for moderately large n.

From this data we would like to derive (a bound on) the differential probability. Let us assume a specific differential occurs k times out of n tries. It is obviously not a good idea to use k/n as a bound on the differential probability. Most possible differences occur 0 times, but we should not assume that they have a 0 probability. If we knew the distribution of the probability we could give some meaningful bound, for example saying that the probability is less than x with a confidence level of 1 %. However, in our case we do not know the distribution of the differential probabilities, and it would be dangerous to assume one. We can, however, reverse the process.

Let us assume that a specific differential has a probability p. If we try the input difference n times, we expect to find this differential around $p \cdot n$ times. The number of times this differential is actually observed is binomially distributed. Let X be a stochastic variable that represents the number of times the differential is observed. We have

$$\Pr(X = k) = \binom{n}{k}(1-p)^{n-k}p^k \qquad k = 0, \ldots, n$$

From this distribution we can derive a bound on the lower tail of the binomial distribution [Fer98]:

$$\Pr(X \leq k) < \Pr(X = k)\frac{p(n-k+1)}{(p \cdot n - k) + p}$$

for $k \leq p \cdot n$. Given a probability p for the differential we can say that we have an unlikely event if the differential occurs k times and $\Pr(X \leq k) < \gamma$ where γ would be a small number. This is a normal test for statistical significance.

We use the following rule to derive a bound on a differential that occurs k out of n times. We use a probability p such that $\Pr(X \leq k) < \gamma$ for some global parameter γ (typical values for γ are 0.05 or 0.01). Of course, we try to choose p as low as possible given this condition. This will overestimate p for most differentials, but underestimate the actual p in a few cases.

We ran these simulations for all input differences with a low enough number of active S-boxes. For every differential that we tried we estimated the probability using this rule. For each set of differentials with the same input and output characterisation we computed the maximum estimated probability. For each differential there is a small chance that we underestimates the probability. However, it is far less likely that we underestimates the maximum probability of a set of differentials. For our maximum to be too low we have to underestimated the probability of the most likely differential. This by itself is rather unlikely. Not only that, we cannot significantly overestimate the probability of any differential with a probability close to the most likely differential. We therefore feel confident that these approximations are reasonable, and that they most likely will result in our overestimating the actual differential probability quite significantly.

For differentials with too many active S-boxes (for which we did not run the simulations) we simply use an upper bound of 1 on the probability of a differential of G.

To improve efficiency we generate our input differences using a straightforward structure. This improves our performance and allows us to increase the number of samples that we make. However, the differentials that we try are no longer independent of each other. We have observed that the use of structures significantly increases the peaks in the bounds. The smaller the structures that we use, the lower the maximum probability bound tends to be. Therefore, we try to reduce the use of structures. We hope our next software version will allow us to eliminate structures altogether.

Apart from these numerical results, we know that certain differential patterns cannot occur. For example, if the input difference is restricted to the first input word of the G, then the output difference must have active bits in both output words. Similarly, if the input difference is restricted to the second input word of G, then the output difference must have active bits in both halves, except when the output of the MDS matrix has a difference of 0x80000000. In this special case, we know that all four S-boxes in this half must be active (otherwise more than 1 byte in the output of the MDS matrix must change). Our software generates all these impossible differential patterns and sets the differential probabilities of the associated sets to zero.

3.2 Differentials of F

Given the results from the last section we can now create a table of upper bounds on the differential probabilities of differentials of F. For each set of differentials we know how many active S-boxes there are. Let σ be the maximum probability of a differential of an S-box. We can now bound the probability of each set of differentials by multiplying the bounds that we found in the previous section on the set by the proper power of σ.

The value σ can be set in various ways. We know that most S-boxes have a best differential probability of 12/256. For the time being we will use this value for σ. Other values, especially larger ones, will be discussed later.

4 Differentials of the round function

Once we have derived bounds on the differentials of F we can do the same for the round function. The differential pattern at the start of the round is characterised by 16 bits, each bit indicates whether the differential pattern in the corresponding byte is nonzero. Given the characterisation of the differential pattern at the input of the round, we know exactly which S-boxes are active. We can generate a list of all suitable differential patterns of F with their associated probability bound. Each of these differentials is combined with the other half of the input differential using the rotate and XOR operations. Each choice of F differential set leads to several possible output differential patterns as the rotate and XORs can lead to different output characterisations depending on the exact differential.

For example, let us look at a differential pattern of 0110 in a 32-bit word. This pattern indicates that only the middle two bytes of the 4-byte word contain active differential bits. After a left rotation, the possible output differential patterns are: 0100, 0110, 1010, 1100, and 1110. XORing two differential patterns can similarly lead to a list of possible results. If we XOR two words, one with a differential pattern of 0101 and one with a differential pattern of 0011, then the possible result patterns are 0110 and 0111.

For each input differential pattern, we can go through all possible F differential patterns, and generate all possible output patterns that can arise. For each possible output pattern we keep track of the largest upper bound that we generate this way. This produces an upper bound for each of the 2^{32} possible input/output differential patterns of the round function.

5 Multi-round patterns

The simplest way of generating multi-round patterns would be to use the list of 2^{32} possible round patterns and a standard search algorithm. We use an algorithm that is somewhat more efficient than that. There are 2^{16} possible differential patterns after r rounds, as each of the 16 data bytes can have a zero or nonzero difference. For each of the 2^{16} possible patterns we store an upper bound on the probability of any characteristic that has this difference pattern after r rounds. Furthermore, we store the list of differential patterns of F, and a precomputed table of how the rotates and XORs can propagate patterns.

For each difference pattern after $r+1$ rounds we use this data to compute an upper bound on the probability of a differential characteristic that has this pattern after $r+1$ rounds.

Given the output pattern of the round in question, we know the first half of the input pattern. This leaves us with 256 possible differential input patterns, and 256 possible differential patterns of F. Each of the 2^{16} possible combinations is tried to see whether it can yield the required output differential pattern. The process can be speeded up by traversing either the F output patterns or the input patterns in decreasing order of probability and using some simple cut-off logic.

6 Results

The results depend on the parameters used to estimate the differential probabilities of G, and the values of γ and σ.

Our current results use $n = 2^{11}$ tries for all differentials with 1 active S-box, and $n = 2^8$ tries for all differentials with 2 active S-boxes. The structure size is 8 and 16 respectively. We use $\gamma = 0.05$, and $\sigma = 12/256$. The full Twofish cipher has 16 rounds. We assume that an adversary can somehow bypass the first round, and can mount a 3R-attack. We thus look at the best 12-round differential characteristic.

With these parameters we found an upper bound on a 12-round differential characteristic of $2^{-102.8}$. This puts a differential attack against Twofish well outside the practical realm.

This upper bound is pessimistic in the following areas:

- The best differential pattern used 3 active S-boxes in 4 of the 12 rounds. The probability of passing a differential with 3 active S-boxes through G is currently taken to be 1. This is clearly overly optimistic, especially since the differential pattern used has both a low input and a low output weight. We believe that extending our simulations to all differentials with 3 active S-boxes will yield a significant further reduction in probability.

- Many of the rounds in the best differential pattern use fancy transformations of the difference pattern by the rotations. This is to be expected of our algorithm, but any non-trivial transformation poses serious restrictions on the actual difference patterns of that word. This makes it much less likely that our upper bound can actually be approached by an actual differential.

- Our estimates are based on the maximum probability of groups of differentials. It is not clear at all that there exists a differential that has a probability that even approaches our upper bound.

7 Other problems for the attacker

To create an attack, the attacker has to choose a specific differential characteristic. That characteristic uses certain specific differences of each of the S-boxes. To get anywhere near the bound all of these differences need to have a probability close to our σ. We chose σ equal to the probability of the best differential of most S-boxes. However, a specific differential will not have the same probability under all keys. If the S-box keys are not known, the attacker has two options. First, he can guess the S-box key bits, and construct a differential characteristic based on that assumption. To achieve good differential probabilities in enough S-boxes, he will have to guess the keys of at least 2 S-boxes (between 32 and 64 bits depending on the key size). Alternatively he can try to find a differential that works for all keys. As we saw in section 2 this leads to very low differential probabilities.

8 Best S-box differential

We use $\sigma = 12/256$ while we know that there are keys for which the best S-box differential has probability 18/256 for a 128-bit key, and even higher for larger key sizes. However, those higher probabilities only occur for a small subset of the keys. We need to address the question how much we are willing to pay in the size of the keyspace the attack is effective on to get a higher probability. If we have an attack that works for 2^{-28} of the key space, how much more efficient should the attack be before it is a better choice for the attacker?

The most natural way is to look at the expected work for the attacker before recovering a single key. We assume that there are enough keys to attack, and optimise the attacker's strategy to find any one key with the least amount of work. This is a reasonable way of looking at the attacker's problem. After all, we know there is an attack that is effective on a

subset of 2^{-64} of the keys with 2^{64} work: a simple exhaustive search of the any subset of that size will do. With such an brute-force attack on a subset of the keys, the expected amount of work before a key is found remains the same.

Let us now look at our Twofish differentials. Suppose we want to use a differential of S-box 0 with probability 14/256; this is possible for about 1 in 8 of all possible keys. We have restricted ourselves to 1/8th of the set of keys, so the workload of our attack should be reduced by at least a factor of 8 for this to be worthwhile. We ran our search for the best differential characteristic pattern again where S-box 0 had a best differential probability of 14/256. The resulting differential probability was 4.6 times higher than the result with $\sigma = 12/256$. Is this worth it?

Let us assume a differential has a probability that depends on the key. We have a list of (p_i, k_i) where the probability of the differential is at most p_i for a fraction k_i of all keys. The expected workload of the attacker to get a single right pair is $1/p_i$ for a fraction k_i of the key space, and thus $\sigma k_i/p_i$ when taken over all keys. The workload is at least $\max k_i/p_i$. This corresponds to the workload of an attack with a differential with probability $\min p_i/k_i$. In our situation the minimum occurs when we use the S-box approximation with probability 12/256. (Using the figures from table 3 in [SK+98], we find that p_i/k_i reaches its minimum at $p_i = 12/256$.) As we currently ignore the $1/k_i$ term, the actual 'effective' probability of a real differential is lower than the bound that we have derived.

We conclude that using a higher probability than 12/256 for an S-box approximation is not worth the loss in key space on which the approximation holds. Thus the bounds we presented earlier hold, and are in fact pessimistic.

9 Other variants

As an experiment we ran the same analysis for Twofish with the 1-bit rotations removed. This makes our approximations match the behaviour of the differential much better. Our results give an upper bound on the probability of a 12-round differential characteristic of $2^{-104.1}$.

This bound is not much better than we have for full Twofish. However, the sequence of differential patterns that achieves this bound uses far more approximations of F that have 3 active S-boxes. In all cases it uses differential characteristics of G that have 3 active input bytes and only a few active output bytes. In practice, such differentials of G will have a far lower probability that the upper bound of 1 that we currently use. Therefore, we expect that our bound can be improved to beyond 2^{-128}.

We have no reason to believe that the 1-bit rotations make Twofish stronger against a differential attack. They were conceived to break up the byte-level structure, but they do not require a separate approximation or increase the avalanche effect of the cipher. We think it is unlikely that the full Twofish has a differential characteristic that is significantly more likely than the version without rotations.

10 Further work

We will continue our analysis work to improve the bound and our understanding of the intricacies of Twofish. We have several areas that we plan to improve.

10.1 Improved G differential estimates

An obvious way to improve our overall bound is to improve our bounds on the differentials of G. We hope to be able to do this in the near future. A larger sample-size will improve the accuracy of our estimates. Extending our computations to differentials of G with 3 active S-boxes should give a great improvement.

10.2 More accurate patterns

Our current pattern-representation is somewhat coarse. We group differentials only by which bytes contain active bits. Apart from the first and final round, all internal differential patterns in our best result have at most 4 active bytes (out of 16). A more fine-grained grouping of the differentials could lead to a better upper bound.

For example, there are only a few G differences with 1 active input byte that have a relatively high probability. Instead of grouping these into the sets, we could treat them separately. This would ensure that our algorithm doesn't magically transform the output of the high-probability difference pattern to one with fewer active bytes by the rotation. The characterisation could be extended with special cases for differences that have only a few active nibbles. We expect that this will result in more S-boxes being

needed for a full differential characteristic, and thus a lower bound.

10.3 Improved treatment of S-box differentials

There is still room to improve our approximations of the S-boxes. We can, for example, compute the best differential approximation for each of the output differences separately. This can then be combined with the analysis of G to get a better bound on differentials of F.

The data on the best S-box differentials in [SK+98] is merged for all the S-boxes. We plan to test the differentials again and collect information for each S-box separately.

We will also improve our handling of the variation of differential probability over the key-space. This will also result in a better bound.

10.4 Additive differentials

We would like to take a closer look at additive differentials modulo 2^{32}. Although we do not expect these to be more useful, it would be nice to derive some bound in that case too.

11 Conclusion

For practical purposes, Twofish is immune to differential cryptanalysis. We have shown that any 12-round differential has a probability of at most $2^{-102.8}$. This bound is far from hard, and we expect that any real differential has a much smaller probability.

The Twofish structure is not easy to analyse. The mixing of various operations makes it hard to give a clean analysis and forces us to use approximation techniques. Some aspects, such as the rotates, make the analysis a lot harder and forces us to use less accurate approximations, while there is no a priori reason to assume that the rotations would have any significant influence on the differential probabilities.

One can argue whether a cipher with a structure that is easier to analyse would be preferable. On the one hand, a structure that allows easier analysis makes it easier to rule out certain attacks. On the other hand, the very structure that makes it easy to analyse might be used in a future attack. Although differential attacks were obviously considered during the design, Twofish was not specifically strengthened against differential attacks, or designed to allow a simple upper bound on differential probabilities to be derived. This is a result of the design philosophy of Twofish. It was not optimised specifically against known attacks; it is a conservative design that tries to resist both known and unknown attacks.

12 Acknowledgements

I would like to thank the other members of the Twofish team: Bruce Schneier, John Kelsey, Doug Whiting, David Wagner and Chris Hall. They encouraged this work and provided many valuable comments.

References

[SK+98] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, and N. Ferguson, "Twofish: A 128-bit Block Cipher", AES submission, june 1998, `http://www.counterpane.com/twofish.html`

[Fer98] N. Ferguson, "Bounds on the tail of binomial distributions", research notes, 1998.