The Boomerang Attack

David Wagner
U.C. Berkeley
daw@cs.berkeley.edu

Abstract. This paper describes a new differential-style attack, which we call the boomerang attack. This attack has several interesting applications. First, we disprove the oft-repeated claim that eliminating all high-probability differentials for the whole cipher is sufficient to guarantee security against differential attacks. Second, we show how to break COCONUT98, a cipher designed using decorrelation techniques to ensure provable security against differential attacks, with an advanced differential-style attack that needs just 2^{16} adaptively chosen texts. Also, to illustrate the power of boomerang techniques, we give new attacks on Khufu-16, FEAL-6, and 16 rounds of CAST-256.

1 Introduction

One of the most powerful cryptanalytic techniques known in the open literature is differential cryptanalysis [BS93]. Differential analysis has been used to break many published ciphers. It is understandable, then, that block cipher designers are typically quite anxious to ensure security against differential style attacks.

The usual design procedure goes something like this. The algorithm designer obtains somehow an upper bound p on the probability of any differential characteristic for the cipher. Then the designer invokes an oft-repeated "folk theorem" to justify that any successful differential attack will require at least $1/p$ texts to break the cipher, which is supposed to allow us to conclude that the cipher is safe from differential attacks.

Unfortunately, this folk theorem is wrong. We exhibit an attack—which we call the boomerang attack—that can allow an adversary to beat the $1/p$ bound in some cases[1]. In particular, if the best characteristic for half of the rounds of the cipher has probability q, then the boomerang attack can be used in a successful attack needing $O(q^{-4})$ chosen texts. In some cases, we may have $q^{-4} \ll p^{-1}$, in which case the boomerang attack allows one to beat the folk theorem's bound. Also, boomerang attacks sometimes allow for a more extensive use of structures than is available in conventional differential attacks, which makes boomerang techniques more effective than the preceding discussion might suggest.

[1] Note that Biham *et al.*'s impossible differentials [BBS98,BBS99] also disprove the folk theorem. They show that if one can find a differential of sufficiently *low* probability, the cipher can be broken. However, the boomerang attack in fact lets us make an sharper statement: even if no differential for the whole cipher has probability that is too high or too low, the cipher might still be vulnerable to differential-style attacks.

L. Knudsen (Ed.): FSE'99, LNCS 1636, pp. 156–170, 1999.

Cipher	(Rounds)	Our Attack	
		Data Complexity	Time Complexity
COCONUT98	(8)	2^{16} CP	2^{38}
Khufu	(16)	2^{18} CP	2^{18}
CAST-256	(16)	$2^{49.3}$ KP	$2^{49.3}$
FEAL	(6)	4 CP	-

KP — known-plaintext, CP — adaptive chosen-plaintext/ciphertext.

Table 1. Summary of our attacks.

We give a surprisingly sharp example of this possibility in Sections 3–5 below, where we show how to break COCONUT98 [V98] with just 2^{16} chosen texts and 2^{38} work, despite a proof that the best characteristic for the whole cipher must have probability $p \approx 2^{-64}$. Our attack makes crucial use of a characteristic for half of the cipher with probability $q \approx 2^{-4}$. This shows that the folk theorem can fail spectacularly, even for real-world ciphers.

We also extend the boomerang attack to use techniques from truncated differential analysis (see Section 6). As a result, we are able to analyze ciphers which admit good truncated differentials. In Section 7 we show how to break 16 rounds of Khufu with 2^{18} adaptive chosen plaintexts and ciphertexts and very little work. We also consider CAST-256 in Section 9, where we show how to break 16 rounds with $2^{49.3}$ known texts[2]. Section 9 also briefly sketches the inside-out attack, a dual to the boomerang attack. Finally, Section 10 discusses some related work, and Section 11 concludes the paper. See Table 1 for our table of results.

2 The Boomerang Attack: A Generic View

The boomerang attack is a differential attack that attempts to generate a quartet structure at an intermediate value halfway through the cipher.

The attack considers four plaintexts P, P', Q, Q', along with their respective ciphertexts C, C', D, D'; we will defer describing how these are generated until later. Let $E(\cdot)$ represent the encryption operation, and decompose the cipher into $E = E_1 \circ E_0$, where E_0 represents the first half of the cipher and E_1 represents the last half. We will use a differential characteristic, call it $\Delta \to \Delta^*$, for E_0, as well as a characteristic $\nabla \to \nabla^*$ for E_1^{-1}.

We want to cover the pair P, P' with the characteristic for E_0, and to cover the pairs P, Q and P', Q' with the characteristic for E_1^{-1}. Then (we claim) the pair Q, Q' is perfectly set up to use the characteristic $\Delta^* \to \Delta$ for E_0^{-1}.

[2] See also Appendix B, where we show that CAST-256 would be much weaker if the round ordering was reversed: in particular, boomerang attacks would be able to break 24 rounds of this variant with $2^{48.5}$ chosen texts. Please note that this 24-round boomerang attack does not apply to the real CAST-256 AES proposal.

Let's examine why this is so. Consider the intermediate value after half of the rounds. When the previous three characteristics hold, we have

$$\begin{aligned}E_0(Q) \oplus E_0(Q') &= E_0(P) \oplus E_0(P') \oplus E_0(P) \oplus E_0(Q) \oplus E_0(P') \oplus E_0(Q') \\ &= E_0(P) \oplus E_0(P') \oplus E_1^{-1}(C) \oplus E_1^{-1}(D) \oplus E_1^{-1}(C') \oplus E_1^{-1}(D') \\ &= \varDelta^* \oplus \nabla^* \oplus \nabla^* = \varDelta^*,\end{aligned}$$

Note that this is exactly the condition required to start the characteristic $\varDelta^* \to \varDelta$ for the inverse of the first half of the cipher. When this characteristic also holds, we will have the same difference in the plaintexts Q, Q' as found in the original plaintexts P, P'. This is why we call it the boomerang attack: when you send it properly, it always comes back to you.

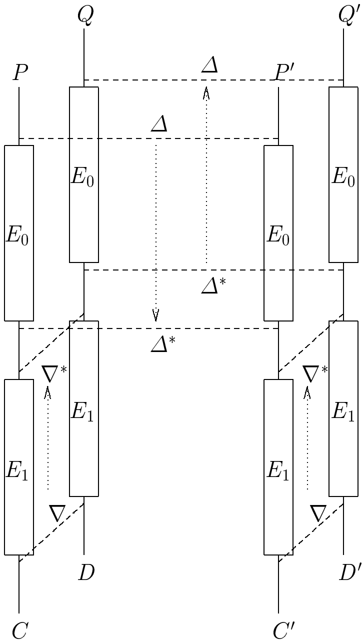

Fig. 1. A schematic of the basic boomerang attack.

We define a right quartet as one where all four characteristics hold simultaneously. The only remaining issue is how to choose the texts so they have the right differences. We suggest generating $P' = P \oplus \varDelta$, and getting the encryptions C, C' of P, P' with two chosen-plaintext queries. Then we generate D, D' as $D = C \oplus \nabla$ and $D' = C' \oplus \nabla$. Finally we decrypt D, D' to obtain the plaintexts Q, Q' with two adaptive chosen-ciphertext queries. See Figure 1 for a pictorial depiction of the basic boomerang attack.

In the remainder of the paper, we consider several concrete attacks using the boomerang attack.

3 The COCONUT98 Algorithm

The COCONUT98 cipher [V98] may be of special interest to some readers because of its reliance on the recently-developed theory of decorrelation techniques for block cipher design [V97,V98,V98b,GG+98]. Using decorrelation techniques, [V98] proves that the full COCONUT98 cipher admits no good differential characteristics. Despite this fact, we observe that there are differential characteristics of very high probability for half of the cipher, and we make extensive use of these characteristics in our attack. This suggests that the decorrelation design technique may fail to provide security against advanced differential attacks in some cases if extra care is not taken. This is not to suggest that the decorrelation approach is fundamentally flawed—indeed, decorrelation theory seems like a very useful tool for the cipher designer—but rather that the theoretical results must be interpreted with caution.

We briefly recount the description of the COCONUT98 algorithm. COCONUT98 uses a 256-bit key $K = (K_1, \ldots, K_8)$. The key schedule generates eight round subkeys $k_1, \ldots, k_8$ as

i	1	2	3	4
k_i	K_1	$K_1 \oplus K_3$	$K_1 \oplus K_3 \oplus K_4$	$K_1 \oplus K_4$

i	5	6	7	8
k_i	K_2	$K_2 \oplus K_3$	$K_2 \oplus K_3 \oplus K_4$	$K_2 \oplus K_4$

The last four key words are used to build a decorrelation module

$$M(xy) = (xy \oplus K_5K_6) \times K_7K_8 \bmod \mathrm{GF}(2^{64})$$

where concatenation of symbols (e.g. xy) represents the concatenation of their values as bitstrings.

Next, we build a Feistel network as follows. Let

$$\begin{aligned}\phi(x) &= x + 256 \cdot S(x \bmod 256) \bmod 2^{32} \\ F_i((x,y)) &= (y, x \oplus \phi(ROL_{11}(\phi(y \oplus k_i)) + c \bmod 2^{32})) \\ \Psi_i &= F_{4i+4} \circ F_{4i+3} \circ F_{4i+2} \circ F_{4i+1}\end{aligned}$$

where $ROL_{11}(\cdot)$ represents a left rotation by 11 bits, c is a public 32-bit constant, and $S : \mathbf{Z}_2^8 \to \mathbf{Z}_2^{24}$ is a fixed S-box.

With this notation, COCONUT98 is defined as $\Psi_1 \circ M \circ \Psi_0$. In other words, COCONUT98 consists of four Feistel rounds with subkeys $k_1, \ldots, k_4$, followed by an evaluation of the decorrelation module M, and finally four more Feistel rounds with subkeys $k_5, \ldots, k_8$.

4 Differential Characteristics for COCONUT98

This section discusses the differential characteristics of COCONUT98. In the following discussion, let $e_j = 2^j$ be the 32-bit XOR difference with just the j-th bit flipped. (Subscripts are taken modulo 32, for convenience in modeling the $ROL(\cdot, 11)$ operation.)

We note that the Feistel rounds of COCONUT98 admit very good differential characteristics. The main observation is that $e_j \to e_{j+11}$ by the Feistel function with probability 1/2 when $j \in J = \{8, 9, \ldots, 19, 20, 29, 30, 31\}$.[3] Similarly, $e_j \oplus e_k \to e_{j+11} \oplus e_{k+11}$ with probability 1/4 when $j, k \in J$ $(j \neq k)$.

Using this idea, we can build many good characteristics for four rounds of COCONUT98. For example, the characteristic

$$(e_{19}, e_{18} \oplus e_8) \to (e_{18} \oplus e_8, e_{29}) \to (e_{29}, e_{18}) \to (e_{18}, 0) \to (0, e_{18})$$

for Ψ has probability $0.83 \cdot 2^{-4} \approx 2^{-4.3}$. Of course, by symmetry we also get corresponding backwards characteristics for decryption through four Feistel rounds.

This suggests that we ought to try to find some way to take advantage of these high-probability characteristics for the half-cipher in our analysis. However, the task is not so easy as it might first look. If we try to mount a traditional differential attack on the whole cipher, the decorrelation module M will immediately cause serious difficulties. When the key words K_7, K_8 are unknown, it is very difficult to push any differential characteristic through M. More precisely, every differential $\delta \to \delta^*$ for M with $\delta, \delta^* \neq 0$ has average probability $1/(2^{64}-1)$, where the probability is averaged over all possible key values. In short, the decorrelation module prevents us from pushing a differential characteristic past M.

This is where the boomerang attack comes in handy: the boomerang quartet property allows us to control the effect of the decorrelation module in the middle.

The crucial idea which lets the attack work is that M is affine, and thus for any fixed key there are excellent characteristics $\nabla^* \to M^{-1}(\nabla^*)$ of probability 1 for M^{-1}. Take $E_0 = \Psi_0$ and $E_1 = \Psi_1 \circ M$. Then if $\nabla \to \nabla^*$ is a good characteristic for Ψ_1^{-1} we will obtain a good characteristic $\nabla \to M^{-1}(\nabla^*)$ for E_1^{-1}. It does not matter that $M^{-1}(\nabla)$ is unknown to the attacker; the crucial property is that it depends only on the key (and not on the values of the ciphertexts).

Let us estimate the success probability for this technique. We need two characteristics for Ψ_0, and two for Ψ_1^{-1}, to hold. Thus, a simple estimate at the probability p of success is

$$p \geq \Pr[\Delta \to \Delta^* \text{ by } \Psi_0]^2 \, \Pr[\nabla \to \nabla^* \text{ by } \Psi_1^{-1}]^2$$

[3] At first glance, it might appear that the probability is 1/8, because there are three additions in the F function and thus three carry bits to control. However, the three carries are not independent, and in fact we can handle three carries as easily as one by noting that $x \mapsto (x + a \bmod 2^{32}) + b \bmod 2^{32}$ (two carries) is equivalent to $x \mapsto x + c \bmod 2^{32}$ (one carry) where $c = a + b$.

The rotate does not destroy this property, so long as we avoid the most significant bits, which explains our choice of J. Empirically, the probabilities are 0.47, 0.44, 0.38 for $j = 18, 19, 20$ and 0.47, 0.44 for $j = 29, 30$. For other values of j, the probability is very close to 1/2.

where $\varDelta, \varDelta^*, \nabla, \nabla^*$ may be chosen arbitrarily by the attacker to maximize p.

It turns out that this estimate can be refined a bit. We note that the same attack works even if we do not predict the exact value of ∇^* ahead of time, but instead merely require that the difference after decrypting by $\varPsi_1$ is the same in the two pairs P, Q and P', Q'. A similar observation also holds for $\varDelta^*$. Therefore, we may sum over all values for $\varDelta^*, \nabla^*$, to obtain

$$p \approx \sum_{\varDelta^*} \Pr[\varDelta \to \varDelta^* \text{ by } \varPsi_0]^2 \cdot \sum_{\nabla^*} \Pr[\nabla \to \nabla^* \text{ by } \varPsi_1^{-1}]^2.$$

For COCONUT98, this can be used to significantly increase the probability of attack. Empirically, we find that $\varDelta = \nabla = (e_{10}, e_{31})$ provides $p \approx 0.023 \cdot 0.023 \approx 1/1900$.

5 The Basic Boomerang Attack on COCONUT98

Next we show how to use the quartet property established above to mount a practical attack on COCONUT98. We use a 1-R attack, so the criterion for success is that $Q \oplus Q' = (?, e_{31})$ where ? represents an arbitrary word. This improves the success probability p by a factor of two, to 1/950.

It is immediately clear from this discussion that COCONUT98 can be easily distinguished from an ideal cipher with at most about $950 \cdot 4 = 3800$ adaptive chosen plaintext/ciphertext queries. However, we aim for more: a key-recovery attack.

The key-recovery attack proceeds along relatively standard lines. In about $16 \cdot 950$ trials requiring $16 \cdot 950 \cdot 4$ adaptive chosen plaintext/ciphertext queries, we generate about 16 useful quartets. Note that the signal-to-noise is extremely high, so we should be able to filter out all wrong quartets very effectively.

First, we recover K_1. We guess K_1, and peel off the first round. We use the fact that if P, P', Q, Q' form a quartet with the property above, then the XOR difference after one round of encryption must be $(e_{31}, 0)$ for both the P, P' pair and the Q, Q' pair. This condition holds for 1/2 of the wrong key values. Therefore each quartet gives one bit of information on K_1 from the P, P' pair and another bit of information from the Q, Q' pair. With 16 useful quartets, we expect K_1 to be identified uniquely.

Next, we recover $K_2 \oplus K_4$ by decrypting up one round and examining the XOR difference in the C, D pair and in the C', D' pair. The details are very similar to those used to learn K_1.

This allows us to peel off the first and last rounds of the cipher. Then we repeat the attack on the reduced cipher. For instance, we can use about $8 \cdot 144 \cdot 4$ more adaptive chosen plaintext/ciphertext queries to generate about 8 useful quartets for the reduced cipher if we use the same settings for $\varDelta, \nabla$, since then the success probability p increases to about 1/144. Using these 8 useful quartets for the reduced cipher we learn K_3; and we repeat the attack iteratively until the entire key is known.

In all, the complexity of the attack is about $16 \cdot 950 \cdot 4 + 8 \cdot 144 \cdot 4 + \ldots \approx 2^{16}$ adaptive chosen plaintext/ciphertext queries. The attack requires $8 \cdot 2 \cdot 32 \cdot 2^{32} = 2^{41}$ offline computations of the F function, which is work comparable to that required for 2^{38} trial encryptions. The attack can also be converted to a known-plaintext attack, but then the complexity increases dramatically to 2^{52} texts.

The best conventional attack on COCONUT98 we could find was a meet-in-the-middle attack that exploits a weakness in the key schedule. However, the meet-in-the-middle attack requires approximately 2^{96} trial encryptions, so our chosen-text boomerang attack compares very favorably to it. See Appendix A for more details on the meet-in-the-middle attack.

Fixing the cipher would require careful changes to its internal design. One possible approach would be to replace the four-round Feistel network Ψ by a transformation with much more strength against differential cryptanalysis (say, 16 rounds instead of 4). Another possible approach is to use a decorrelation module in each round; this seems likely to prevent boomerang-style attacks, and is in fact the approach proposed in the DFC AES submission [GG+98]. (Using just a decorrelation module before the first round and after the last round is not enough—differential-style attacks are still possible.)

It is clear that the mere use of decorrelation techniques is not enough to guarantee security against differential-style attacks. At the same time, although it does not provide the conjectured 2^{64} security level, COCONUT98's decorrelation module does seem to improve the cipher's security. Without a decorrelation module, COCONUT98 would be vulnerable to conventional differential attacks requiring on the order of 2^{8} chosen texts, so in this case the decorrelation module seems to have approximately squared the security level of the base cipher.

6 Extensions to Truncated Differential Analysis

So far we have confined the discussion to conventional differential characteristics, but it seems natural to wonder whether boomerang attacks can also be made to work using truncated differentials. The answer is yes, but there are some difficulties.

The pitfall with extensions to truncated differentials is that

$$\Pr[\Delta \to \Delta^* \text{ by } F] = \Pr[\Delta^* \to \Delta \text{ by } F^{-1}]$$

always holds for conventional differential characteristics, but can fail to hold for truncated characteristics. Note that our analysis in earlier sections assumed that if $\Delta \to \Delta^*$ by the first half of the cipher, then $\Delta^* \to \Delta$ holds with the same probability for the inverse of the first half of the cipher. For truncated differentials, this assumption in general is not correct.

A more accurate formula for the success probability p of a boomerang attack with truncated differentials is

$$p \approx \sum_{w \oplus x \oplus y \oplus z = 0} \Pr[\Delta \to w \text{ by } E_0] \times \Pr[\nabla \to x \text{ by } E_1^{-1}] \times \Pr[\nabla \to y \text{ by } E_1^{-1}] \times \Pr[z \to \Delta \text{ by } E_0^{-1}].$$

This formula is rather unwieldy, but fortunately it can often be simplified substantially to

$$\begin{aligned} p \approx\ & \Pr[\varDelta \to \varDelta^*] \times \Pr[\nabla \to \nabla^*]^2 \times \Pr[\varDelta^* \to \varDelta] \times \\ & \Pr[w \oplus x \oplus y \in \varDelta^* \mid w \in \varDelta^*, x, y \in \nabla^*]. \end{aligned}$$

If the truncated differentials $\varDelta^*, \nabla^*$ are linear (i.e. closed under $\oplus$), as is usually the case, the last term in the formula above is easily computed.

7 Khufu

We describe a boomerang attack that breaks 16 rounds of Khufu [Mer90] with 2^{18} adaptively chosen plaintext/ciphertext queries and a comparable workfactor. This is an improvement over the best previous result, a differential attack on Khufu-16 needing 2^{31}–2^{43} chosen texts (depending on whether one wants a distinguishing or key-recovery attack) [GC94].

In our boomerang attack, we exploit that there are excellent truncated differentials available for both halves of the cipher. For the first half of the cipher, we use

$$\varDelta = (0, 0, 0, a, b, c, d, e) \to (0, 0, 0, a, 0, 0, 0, 0) = \varDelta^*,$$

which holds with probability 2^{-32} in the forward direction and probability 1 in the reverse direction. We will hold a fixed throughout the attack. For the inverse of the last half of the cipher, we use $\nabla = (0, 0, 0, a, 0, 0, 0, 0) \to (0, 0, 0, a, f, g, h, i) = \nabla^*$, which holds with probability 1. Also, due to a careful choice of $\nabla^*, \varDelta^*$, we have $\Pr[w \oplus x \oplus y \in \varDelta^* \mid w \in \varDelta^*, x, y \in \nabla^*] = 1$. Thus 2^{-32} of the quartets chosen according to these differences will form right quartets.

One can use structures to reduce the number of texts needed. Choose a pool of 2^{16} plaintexts (L, R_i) with L held fixed and R_i varying. Also, form another pool of 2^{16} plaintexts as (L', R'_j) where $L' = L \oplus (0, 0, 0, a)$ and R'_j varies. For each ciphertext C obtained by encrypting one of these 2^{17} plaintexts, we decrypt $D = C \oplus \nabla$ to get the plaintext Q. We look for Q, Q' with a difference of $(0, 0, 0, a)$ in the left half of the block; such a pair probably indicates a right quartet. This choice of structures is expected to provide about one right quartet, although one wrong quartet will probably also survive the initial filtering phase.

Once we have a (suggested) right quartet formed by (L, R_i) and (L', R'_j), we can use it to obtain more right quartets at little cost. We form another 2^{10} quartets by choosing $P = (L \oplus (\alpha, \beta, 0, 0), R_i)$, $P' = (L' \oplus (\alpha, \beta, 0, 0), R'_j)$ where α, β take on 2^{10} possible values; C, C', D, D', Q, Q' are generated from P, P' as before. Now each such quartet is guaranteed to be a right quartet (if $(L, R_i), (L', R'_j)$ formed a right quartet) because we have successfully bypassed the first round. Thus, any wrong quartets which survived the earlier filtering phase are easily eliminated. Furthermore, given 2^{10} right quartets we expect to be able to form 2^{10} equations of the form $S_1(x) \oplus S_1(y) = z$ for known values of x, y, z, and this should be sufficient to recover S_1 up to a XOR by a 32-bit constant. Then the 8-round reduced cipher can be broken trivially.

In total, this attack on Khufu-16 requires $2^{18} + 4 \times 2^{10} \approx 2^{18}$ adaptively chosen texts. The workfactor is minimal.

8 FEAL

One can also apply boomerang techniques to FEAL. There are 3-round differential characteristics with probability one [BS93], so we immediately obtain an efficient boomerang attack that distinguishes FEAL-6 from a random permutation with only four adaptive chosen plaintext/ciphertext queries. (This elegant observation is due to Eli Biham [Bih99].)

9 Inside-Out Attacks

In this section, we sketch a description of the "inside-out attack," which may be viewed as a dual to the boomerang attack. The difference is that the boomerang attack works from the outside in while the inside-out attack works from the inside out.

In the inside-out attack, we search for pairs of texts which contain a desired difference Δ at the intermediate value after half the rounds. We hope that the differential $\Delta \to \Delta'$ for E_1 and the differential $\Delta \to \Delta^*$ for E_0^{-1} both hold. In this case, we will have recognizable differences Δ^* and Δ' in the plaintexts and ciphertexts of the pair. If we accumulate enough pairs with the difference Δ halfway through the cipher, we should be able to find at least one right pair where both differentials hold.

To illustrate these ideas in action, we analyze 16 rounds of CAST-256. CAST-256 [Ada98] is a generalized Feistel block cipher, whose simplicity makes it a nice test-bed to explore the properties of generalized Feistel round structures.

We briefly recall the definition of CAST-256 here. The 128-bit block is divided into four 32-bit words, and a Feistel function $F : \mathbf{Z}_2^{32} \to \mathbf{Z}_2^{32}$ is used to update the block. There are two types of rounds, which we shall call "A rounds" and "B rounds" in a choice of terminology inspired by Skipjack. An A round encrypts the input block (w, x, y, z) to $(z, w, x, y \oplus F(z))$, and a B round encrypts to $(x, y, z \oplus F(w), w)$. Note that $A \approx B^{-1}$; by this we mean that the structure of the inverse of a B round is the same as the structure of an A round, not that they are true functional inverses. With this terminology, the CAST-256 cipher structure is defined as $B^{24} \circ A^{24}$, i.e. 24 A rounds followed by 24 B rounds.

The CAST-256 structure admits many nice truncated differentials. In our boomerang attack, we will use $\Delta = (0, 0, 0, a) \to (0, b, c, a) = \Delta'$, which holds with probability 1 for 8 B rounds, and $\Delta = (0, 0, 0, a) \to (0, d, e, a)$, which holds with probability 1 for decrypting though 8 A rounds.

The signal-to-noise ratio of the inside-out attack will be reasonably good, because right pairs can be recognized by a 96-bit filtering condition.

To implement the attack, we collect $2^{49.3}$ known texts encrypted under 16 rounds of CAST-256. By the birthday paradox, we expect to see three right pairs among those texts, which can be readily recognized. (We also expect to get three wrong pairs, but they should be eliminated in the next phase.) Then we search over the last round subkey. Each guess at the 37 key bits entering the last round suggests 2^5 possible values for the 37 key bits entering the next-to-last round; the three right pairs allow us to uniquely recognize the correct values for the last two round subkeys. The first two round subkeys can be recovered by analogous techniques. Finally, the attack may be repeated on the reduced-round cipher.

To sum up, we see how to break 16 rounds of CAST-256 with an inside-out attack that needs just $2^{49.3}$ known texts and very little work. This attack is independent of the definition of F function or key schedule, and depends only on the round structure.

There are two implications of our analysis. First, it indicates that CAST-256 reduced to 16 rounds would not be adequately secure. Since CAST-256 with 48 rounds is 2–2.5 times slower on high-end CPUs than the fastest AES candidates [SK+98], this suggests that CAST-256's security-to-performance ratio may not be as high as some other contenders. On the other hand, security clearly must take precedence over performance, and here our analysis provides some support for the CAST-256 design. We have seen that CAST-256's round ordering is ideally-suited to resist boomerang attacks (see Appendix B), and due to the sheer number of rounds, it seems very hard to extend our inside-out attack to the full cipher.

10 Related Work

The boomerang attack is closely related to many other ideas that have previously occurred in the literature. As a result, there are many different ways to think about the boomerang attack. In this section, we will try to survey the possibilities.

The boomerang attack is related to the differential-linear attack of [HL94]. In a differential-linear attack, one covers E_0 with a truncated differential $\varDelta \to \varDelta^*$, covers E_1^{-1} with a linear approximation $\varGamma \to \varGamma^*$, and finally covers E_1 with a second approximation $\varGamma^* \to \varGamma$; there is also the additional requirement that $\varGamma^* \cdot x$ be constant for all $x \in \varDelta^*$. From this perspective, one could think of the boomerang attack as a "differential-differential" attack (if the reader will indulge a slight abuse of terminology).

A similar observation is that the boomerang attack is closely related to higher-order differential techniques [Lai94,Knu95]. As noted in Section 6, the pairs P, Q, P', Q' don't actually need to follow $\nabla \to \nabla^*$: it is sufficient that $E_1^{-1}(P) \oplus E_1^{-1}(P') \oplus E_1^{-1}(Q) \oplus E_1^{-1}(Q') = 0$, and this may be viewed (in a very approximate sense) as a higher-order differential of order two. In this way, the boomerang attack can be considered as an intermediate step between conventional differential and higher-order differential attacks.

Another precursor of the boomerang attack is the "double-swiping" attack [KSW97], a differential related-key attack on NewDES-1996 that can, in retrospect, be viewed as a boomerang-style attack (with minor adjustments to take advantages of related-key queries, as allowed in [KSW97]'s extended threat model).

One of the interesting features of the boomerang attack is that it is apparently very well-suited to the analysis of ciphers that use asymmetric round functions[4]. Asymmetric round functions can be classified into one of two types: the A round, which has better diffusion in the forward direction than in the reverse direction, and the B round, which has better diffusion in the reverse direction. We note that when the first half of the cipher is built of B-type rounds and the last half is built of A-type rounds, boomerang attacks seem to be especially dangerous because they allow one to probe from both endpoints at the same time.

This supplies some intuition for how the boomerang attack works. It would not be unreasonable to think of the boomerang attack as a differential meet-in-the-middle attack that uses differentials to work from the outside in; the interesting bit is what happens where the differentials "meet" in the middle of the cipher.

One disadvantage of the boomerang attack is that it inherently requires the ability to perform both adaptive chosen-plaintext and adaptive chosen-ciphertext queries at once, a rare requirement to find in a practical attack. We are aware of only two other attacks with this property: (1) the adaptive chosen-plaintext/ciphertext attack on the 3-round Luby-Rackoff cipher, which is also used to good effect in some of Knudsen's work [Knu98] on Luby-Rackoff ciphers with more rounds, and (2) Biham *et. al*'s yo-yo game [BB+98], which is closely related to their more-famous miss-in-the-middle attack [BBS98,BBS99].

The relation between the boomerang attack and the miss-in-the-middle attack is a close and interesting one. It seems that the boomerang attack is little more than a chosen-plaintext/ciphertext version of the miss-in-the-middle attack. In particular, if $\Pr[\varDelta \to \varDelta^*] = \Pr[\nabla \to \nabla^*] = 1$ and $\varDelta^* \cap \nabla^* = \emptyset$, then the same pair of differentials can be used to obtain either a miss-in-the-middle attack (using the impossible differential $\varDelta \to \nabla$) or a boomerang attack.

This paper showed that in some special cases the boomerang attack can improve on the miss-in-the-middle attack, if adaptive chosen plaintext/ciphertext queries are available. However, this seems to be the exception rather than the rule. For several ciphers—including Skipjack and CAST-256—miss-in-the-middle attacks penetrate through more rounds than boomerang attacks [BBS98,BBS99]. Though a thorough comparison of the two types of attacks continues to elude us, we hope that this work will stimulate further research into the interaction between these two attacks.

11 Conclusions

We have described a new way to use differential-style techniques for cryptanalysis of block ciphers. Our attacks can break some ciphers that are immune to ordinary differential cryptanalysis, and can provide a powerful new way to analyze ciphers

[4] See Appendix B for a concrete example of this.

with asymmetrical round structures. To protect against these attacks, cipher designers should ensure that there are no good differentials for the first or last half of their cipher.

12 Acknowledgements

We are grateful to a number of readers whose comments have improved the paper substantially: Serge Vaudenay offered a number of helpful comments on an early version of this paper and allowed us to use his nice one-line summary of the attack; Eli Biham suggested looking at reduced-round versions of FEAL, and pointed out the relevance of the 3-round characteristics of probability 1; Carlisle Adams, Alex Biryukov, and the anonymous reviewers gave helpful feedback on the exposition; and finally, we would like to thank all those patient readers who suggested including Figure 1 in this paper.

References

[Ada98] C. Adams, "The CAST-256 Encryption Algorithm," NIST AES Proposal, Jun 98.

[Ada99] C. Adams, personal communication, Feb 1999.

[BBS98] E. Biham, A. Biryukov, A. Shamir. "Cryptanalysis of Skipjack Reduced to 31 Rounds using Impossible Differentials," *EUROCRYPT'99*, to appear.

[BBS99] E. Biham, A. Biryukov, A. Shamir. "Miss in the Middle Attacks on IDEA, Khufu, and Khafre," this volume.

[BB+98] E. Biham, A. Biryukov, O. Dunkelmann, E. Richardson, A. Shamir, "Initial Observations on the Skipjack Encryption Algorithm," *SAC'98*, Springer-Verlag, 1998.

[Bih99] E. Biham, personal communication, Mar 1999.

[BS93] E. Biham, A. Shamir, *Differential Cryptanalysis of the Data Encryption Standard*, Springer-Verlag, 1993.

[BC+98] C. Burwick, D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S.M. Matyas, L. O'Connor, M. Peyravian, D. Safford, and N. Zunic, "MARS — A Candidate Cipher for AES," NIST AES Proposal, Jun 98.

[GG+98] H. Gilbert, M. Girault, P. Hoogvorst, F. Noilhan, T. Pornin, G. Poupard, J. Stern, S. Vaudenay, "Decorrelated Fast Cipher: an AES Candidate," NIST AES Proposal, Jun 98.

[GC94] H. Gilbert and P. Chauvaud, "A chosen plaintext attack of the 16-round Khufu cryptosystem," *CRYPTO'94*, LNCS 839, Springer-Verlag, 1994.

[GLC98] D. Georgoudis, D. Lerous, and B.S. Chaves, "The 'Frog' Encryption Algorithm," NIST AES Proposal, Jun 98.

[HL94] M. Hellman and S. Langford., "Differential–linear cryptanalysis," *CRYPTO'94*, LNCS 839, Springer-Verlag, 1994.

[KSW97] J. Kelsey, B. Schneier, D. Wagner, "Related-Key Cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA," *ICICS'97*, Springer-Verlag, 1997.

[Knu95] L.R. Knudsen, "Truncated and Higher Order Differentials," *Fast Software Encryption, 2nd International Workshop Proceedings*, Springer-Verlag, 1995.

[Knu98] L. Knudsen, “DEAL—A 128-bit Block Cipher,” NIST AES Proposal, Jun 98.

[Lai94] X. Lai, “Higher Order Derivations and Differential Cryptanalysis,” *Communications and Cryptography: Two Sides of One Tapestry*, Kluwer Academic Publishers, 1994, pp. 227–233.

[Mer90] R. C. Merkle, “Fast Software Encryption Functions”, *CRYPTO’90*, Springer-Verlag, 1990.

[NW97] R. Needham and D. Wheeler, “TEA Extensions,” unpublished manuscript, Mar 1997.

[NSA98] NSA, “Skipjack and KEA algorithm specifications,” May 1998. Available from `http://csrc.ncsl.nist.gov/encryption/skipjack-1.pdf`.

[Saa98] M.-J. Saarinen, “Cryptanalysis of Block Tea,” unpublished manuscript, 20 Oct 1998.

[SK+98] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson, “Performance Comparison of the AES Submissions,” *Second AES Conference*, 1999.

[V97] S. Vaudenay, “A cheap paradigm for block cipher strengthening,” LIENS tech report 97-3, 1997.

[V98] S. Vaudenay, “Provable Security for Block Ciphers by Decorrelation,” *STACS’98*, Springer-Verlag LNCS 1373, 1998.

[V98b] S. Vaudenay, “Feistel Ciphers with L_2-Decorrelation,” *SAC’98*, Springer-Verlag, 1998.

[Yuv97] G. Yuval, “Reinventing the Travois: Encryption/ MAC in 30 ROM Bytes,” *FSE’97*, LNCS 1267, 1997.

A Meet-in-the-Middle Attack on COCONUT98

The very simple key schedule used in COCONUT98 exposes it to meet-in-the-middle attacks. The problem is that there are only 96 bits of entropy in the first four round subkeys, and a similar property holds for the last four round subkeys. Therefore, with just four known texts and about 2^{96} offline work, one can break COCONUT98 using standard meet-in-the-middle techniques[5]. The workfactor of this attack is disappointingly low for a cipher with a 256-bit key.

When the key is chosen non-uniformly, e.g. from a passphrase, this attack can be even more deadly. If we assume a key entropy of 4 bits/byte (probably a gross overestimate for most passphrases), the workfactor of the meet-in-the-middle attack can be reduced to approximately 2^{48} trial encryptions. This is much faster than exhaustive keysearch.

B A CAST-256 Variant

In this section, we consider a simple CAST-256 variant obtained by exchanging the order of the A rounds and the B rounds. (In other words, the variant cipher

[5] Specifically: Obtain four known text pairs P_j, C_j for $j = 1, 2, 3, 4$. Guess K_3, K_4. For each possibility for K_1, store $(\Psi_0(P_0) - \Psi_0(P_1))/(\Psi_0(P_2) - \Psi_0(P_3)), K_1$ in a lookup table keyed on the first component. Finally, for each possibility for K_2, we compute $(\Psi_1^{-1}(C_0) - \Psi_1^{-1}(C_1))/(\Psi_1^{-1}(C_2) - \Psi_1^{-1}(C_3))$ and look for a match in the lookup table.

uses the B rounds first.) The primary contribution is that such a variant can be readily analyzed using boomerang attacks.

Please note that this attack *does not* apply to CAST-256 (only to a variant with a different round structure)[6]. Since the designers of CAST-256 already knew of the need to apply the A rounds first [Ada99], we feel that the variant does injustice to the spirit of the CAST-256 design. We focus on CAST-256 primarily because it makes such a simple, clean platform for analysis of novel round structures. We believe our attack on this CAST-256 variant gives new insights into the properties of various ciphers with generalized Feistel round structures [NSA98,BC+98,GLC98,Yuv97,NW97,Saa98], so we hope the analysis is of independent interest.

The sheer number of rounds makes it hard to mount good attacks on the full 48-round CAST-256. In this section, we show that boomerang attacks with complexity $2^{48.5}$–2^{65} are possible on 24–25 rounds of the variant cipher. These attacks do not appear to extend to the original CAST-256 round ordering, so we believe this provides some additional justification that CAST-256 is using the right round ordering.

A SIMPLE ATTACK ON 24 ROUNDS. We use the truncated differential $\varDelta = (0,0,0,a) \to (b,c,d,a) = \varDelta^*$ for the 12 B rounds (where a may take on any nonzero value, and b,c,d are arbitrary). For the inverse of the last half of the cipher, we use a similar truncated differential: namely, $\nabla = (0,0,0,e) \to (f,g,h,e) = \nabla^*$.

Using the machinery developed in Section 6, the computation of the success probability is straightforward. Both of these truncated differentials have probability 1, and $\varDelta^* \to \varDelta$ has probability 2^{-96}. Finally, we note that

$$\Pr[w \oplus x \oplus y \in \varDelta^* \mid w \in \varDelta^*, x, y \in \nabla^*] = \Pr[a \oplus e \oplus e' \neq 0] = 1 - 2^{-32},$$

so the overall success probability is $p \approx 2^{-96}$.

We start the attack by choosing 2^{32} plaintexts P_i where the first three words are held fixed and the last takes on all 2^{32} possibilities, and we obtain the corresponding ciphertexts C_i. For each such ciphertext C_i, we generate $2^{16.5}$ new ciphertexts $D_{i,j}$ by varying the final word. Then we decrypt each $D_{i,j}$ to obtain the corresponding plaintext $Q_{i,j}$. This gives us 2^{63} choices for P, P' from the pool of plaintexts P_i and another 2^{33} choices for D, D' from the $D_{i,j}$. In all, there will be 2^{96} possible quartets to choose from. About $p \approx 2^{-96}$ of them will form right quartets, so we expect to see one right quartet. The excellent filtering available (we can filter on all 128 bits of $Q \oplus Q'$) allows us to eliminate all the wrong quartets with high probability.

This immediately gives a way to distinguish the 24-round CAST-256 variant from a ideal cipher with $2^{48.5}$ adaptively chosen texts and a low workfactor.

A KEY-RECOVERY ATTACK ON 25 ROUNDS. The same ideas can also be used to develop key recovery attacks. For instance, we can break the 25-round variant obtained by prepending one more B round at the beginning with 2^{65} chosen

[6] See also Section 9, which analyzes 16 rounds of the real CAST-256 cipher (without any re-ordering of the rounds).

texts and a similar amount of work. Due to lack of space, we give only a very brief sketch of the attack: we bypass the first round with structures, and then in the analysis phase we guess the first-round subkey, peel off the first round, and check for the existence of right quartets.

Discussion. It is worth comparing our results to what is attainable with conventional truncated differential cryptanalysis. In the case of this CAST-256 variant, boomerang attacks seem to compare favorably for up to 24 rounds, due to the asymmetric round structure, but for more than 25 rounds conventional techniques are at least as good as the boomerang.

The astute reader will have noticed that our truncated differentials $\Delta \to \Delta^*$ (for the 12 B rounds) and $\nabla^* \to \nabla$ (for the 12 A rounds) can be readily concatenated to obtain a truncated differential $\Delta = (0,0,0,a) \to (0,0,0,a) = \nabla$ of probability 2^{-96} for the entire cipher. The resulting 24-round differential will have probability 2^{-96}, and can be used in a conventional truncated differential attack that distinguishes the 24-round CAST-256 variant from a ideal cipher with 2^{65} (non-adaptive) chosen plaintexts. Note that you can also get a miss-in-the-middle attack on the 24-round variant with the same techniques, since $(0,0,0,a) \to (0,0,0,a')$ is an impossible differential when $a \neq a'$. This gives an attack that uses 2^{65} chosen plaintexts and not much work.

Thus, our 24-round boomerang attack ($2^{48.5}$ adaptive chosen-plaintext and chosen-ciphertext queries) seems better than the conventional truncated differential attack (2^{65} chosen plaintexts) or the miss-in-the-middle attack (2^{65} chosen plaintexts), but it loses its advantage at 25 rounds.

One reason why the boomerang attack succeeds against the CAST-256 variant is that CAST-256 rounds exhibit a definite asymmetry. In both Skipjack and CAST-256, the A rounds have weaker diffusion in the reverse direction than in the forward direction, while the B rounds are stronger in the reverse direction. Thus, the combination of A and B rounds makes conventional differential attacks harder than usual: whether we attack the cipher or the inverse cipher, we will have to push a differential through 12 “strong rounds”. In contrast, the boomerang attack allows us to follow the path of least resistance in both directions, because we cover the B rounds with a differential running in the forward direction and cover the A rounds with a differential running in the reverse direction. This makes the boomerang attack especially well-suited to the analysis of a cascade of B rounds followed by A rounds.

By the same line of reasoning, boomerang techniques would be especially weak at analyzing the real CAST-256 cipher, where the A rounds precede the B rounds. A boomerang attack on CAST-256 would be attacking the cipher at all of its strongest points, and thus boomerang techniques would be a particularly poor tool for analyzing the real CAST-256 round structure.