

The Boomerang Attack on 5 and 6-round Reduced AES*

Alex Biryukov

Katholieke Universiteit Leuven, Dept. ESAT/SCD-COSIC,
Kasteelpark Arenberg 10,
B-3001 Heverlee, Belgium
<http://www.esat.kuleuven.ac.be/~abiryuko/>

Abstract. In this note we study security of 128-bit key 10-round AES against the boomerang attack. We show attacks on AES reduced to 5 and 6 rounds, much faster than the exhaustive key search and twice faster than the “Square” attack of the AES designers. The attacks are structural and apply to other SPN ciphers with incomplete diffusion.

1 Introduction

In this paper we study security of 128-bit key AES [2] against the boomerang attack [4]. The boomerang attack was developed in 1999 after the AES competition was already running. This attack sometimes allows to break more rounds than the conventional differential or linear attacks, especially for the ciphers with few but carefully designed rounds (for example, see an attack on SAFER++ [1]).

In this paper we show attacks on AES reduced to 5 and 6 rounds. Six round attack has complexity of 2^{71} data and steps of analysis (measured in 6-round encryptions). The attack is twice faster than the “Square” attack of the designers of the AES in terms of time complexity which is a dominant factor, but has much higher data complexity. The boomerang attack on AES is less efficient than the partial sum attack [3]. See Table 1 for comparison of our attacks with the previous results on a 128-bit key AES.

2 Boomerang Attack on SPNs with Incomplete Diffusion

Boomerang attack is a chosen plaintext-adaptive chosen ciphertext attack. It is an extension of differential cryptanalysis and works on quartets of data $(P, P'), (Q, Q')$.

* This work was supported in part by the Concerted Research Action (GOA) Mefisto-2000/06 of the Flemish Government and in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The information in this document reflects only the author’s views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

Table 1. Comparison of our results with previous attacks on AES.

Attack	Key size	Rounds	Data ^a	Type ^b	Workload ^c	Memory ^a
Square attack (Daemen-Rijmen'98)	128	5 of 10	2^{11}	CP	2^{40}	2^{11}
Square attack (Daemen-Rijmen'98)	128	6 of 10	2^{32}	CP	2^{72}	2^{32}
Collision (Gilbert-Minier'00)	128	7 of 10	2^{32}	CP	2^{128}	2^{80}
Partial sum (Ferguson <i>et al.</i> '00)	128	6 of 10	$2^{34.6}$	CP	2^{44}	2^{32}
Partial sum (Ferguson <i>et al.</i> '00)	128	7 of 10	$2^{128}-2^{119}$	CP	2^{120}	2^{64}
Imposs. diff. (Biham-Keller'00)	128	5 of 10	$2^{29.5}$	CP	2^{31}	2^{40}
Imposs. diff. (Cheon <i>et al.</i> '01)	128	6 of 10	$2^{91.5}$	CP	2^{122}	2^7
Our Boomerang attack	128	5 of 10	2^{39}	CP/ACC	2^{39}	2^{33}
Our Boomerang attack	128	6 of 10	2^{71}	CP/ACC	2^{71}	2^{33}

^a Expressed in the number of blocks.

^b CP – Chosen Plaintext, ACC – Adaptive Chosen Ciphertext.

^c Expressed in equivalent number of encryptions.

The attack works when encryption $E()$ can be split into $E = E_1 \circ E_0$, where E_0 is weak in encryption direction and E_1 is weak in decryption direction. We refer the reader to [4] for further details.

In this section we present a generic method of breaking five and six round substitution-permutation networks (SPNs) using a boomerang distinguisher. The attacks that we will show will be *structural* in the sense that they will not use specific properties of S-boxes or of the mixing layer, but will use only the fact that diffusion is incomplete (which is the case for many ciphers, including the AES).

We will describe this attack on an example of Rijndael-like cipher with layers of 16, 8x8-bit S-boxes, and RIJNDAEL-like diffusion involving ShiftRows and MixColumns (though exact constants in the MixColumns matrix will be irrelevant to the attack).

The five round attack will be as follows:

1. Prepare a pool of plaintexts $\{P_i\}, i = 0, \dots, 2^{32} - 1$ which have all possible values in four bytes (which will appear in the same column before the MixColumns) and arbitrary constant in the other bytes. Encrypt the pool and obtain a pool of 2^{32} ciphertexts $\{C_i\}$.
2. Construct a pool of modified ciphertexts: $D_i = C_i \oplus \nabla$, where ∇ is a fixed non-zero difference with only one active S-box (for example, a non-zero difference in the first byte and zero difference in 15 other bytes).
3. Decrypt the pool $\{D_i\}$ to obtain a pool $\{Q_i\}$ of 2^{32} new plaintexts.
4. Sort the pool $\{Q_i\}$ by the bytes corresponding to eight inactive S-boxes. Pick only those pairs Q_i, Q_j which have zero difference in these 8 bytes. If none found go to step 1.
5. For each of the quartets P_i, P_j, Q_i, Q_j that pass step 4, guess the 32-bit key value that enters the four S-boxes corresponding to non-constant bytes.

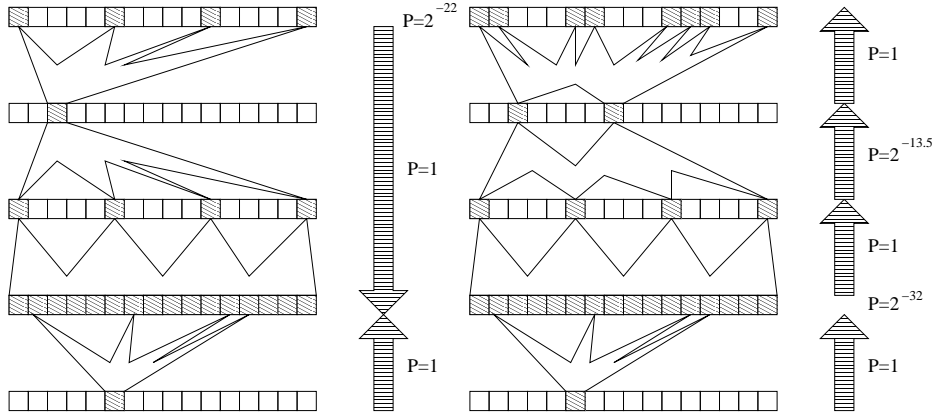


Fig. 1. Schematic description of a boomerang quartet for Rijndael reduced to five rounds.

Using the guessed key value partially encrypt one round and check that resulting difference is in a single active S-box, which is a 22-bit filtering condition for each pair (P_i, P_j) and (Q_i, Q_j) . This gives a 44-bit condition in total for both sides of the boomerang in the case of common 4-tuples of active S-boxes. However with probability half we will have no common 4-tuples, i.e. all the twelve active S-boxes (4 from the (P_i, P_j) pair and 8 from the (Q_i, Q_j) pair) non-overlapping. We will then pick key-candidates that are suggested at least twice.

See Figure 1 for a schematic description of the boomerang distinguisher used in this 5-round attack. The rectangles in the figure denote the layers of S-boxes, and the gray squares indicate the active S-boxes. Arrows represent the cost of pattern propagation in terms of probability.

Complexity of the 5-round attack Analysis of the complexity of the attack described above is as follows: Each pool of 2^{32} texts contains 2^{63} pairs with difference in the four relevant bytes. From these pairs $2^{63}/2^{22} = 2^{41}$ will have a single active S-box after one round¹. After the second round we will have four active S-boxes. After the 3rd round all bytes will be active. From the bottom up direction we will have one round crossed with probability one, with a truncated differential that starts with one active S-box and ends with four active S-boxes. At this point we need that after the next S-box layer the difference in these four active S-boxes would be the same. This happens with probability 2^{-32} . Then we can switch to the last face of the boomerang, where the effect of the mixing of the 3rd round can then be undone with probability one and we will get four active S-boxes after the S-box layer of the 3rd round. We will have to pay $2^{-13.5}$

¹ The chance of going from 4 active S-boxes to one is $4 \cdot 2^8 / 2^{32} = 2^{-22}$, since we do not care about the location of the single active S-box.

in probability for the four active S-box difference to turn into two active S-box difference after this S-box layer and the MixColumns which is just above it. From that point we let our truncated differential run freely with probability 1. As a result we will obtain a new pair of plaintexts with some difference in eight bytes and zero difference in the other eight bytes. This is our $64 - \log(6)$ -bit filtering condition for the good boomerang quartets (the $-\log(6)$ appears since we do not fix the places of the two active S-boxes).

We pick about 2^6 pools in which we will have $2^{41} \cdot 2^6 \cdot 2^{-13.5} \cdot 2^{-32} \approx 3$ good boomerangs returning back. The average amount of false quartets which satisfy our initial filtering condition is $2^{63} \cdot 2^6 \cdot 2^{-64} \cdot 6 = 192$.

In the simplest case when the boomerang returns in the same four bytes as it was sent we perform a guess of the 32-bit key and check it against two sides of the boomerang P_i, P_j and Q_i, Q_j whether in both cases it leads to a single active S-box after one round. This gives a 44-bit filtration condition which leaves only the correct key guess with probability $1 - 2^{-12}$. However with probability 1/2 it may happen that for the two boomerang pairs active 4-tuples of the output pair (Q_i, Q_j) will be different from those of the input pair (P_i, P_j) . In this case we independently guess 32-bits of the key corresponding to the input four bytes for each pair and leave only those keys that lead to a single active S-box after the first round. We expect at least two good boomerang quartets in our pools and thus the correct key would be counted at least twice. Note that we have about 100 noisy pairs (those with non-overlapping 4-tuples) each of which suggests about $4 \cdot 2^8$ candidates for the 32-bit key. That means that we may have a few wrong keys suggested due to the birthday collisions together with the correct key suggested by the good boomerangs. The same analysis can be performed in parallel on another 4-tuple to produce few candidates for another 32-bit part of the key. Knowing a few candidates for at least half of the first subkey we can repeat the attack with much less data and smaller complexity to achieve the full key-recovery.

Total complexity of this 5-round attack is 2^{38} chosen plaintext/adaptive chosen ciphertext queries and 2^{38} time steps which mainly would be spent on encrypting and sorting the data. The memory required by the attack is 2^{32} blocks or 2^{36} bytes.

Extension to 6-rounds of AES The attack described above can be extended by one round at the bottom at the cost of guessing 32-bits of the key of the 6th round. We double the number of pools from 2^6 to 2^7 to get 4-6 good quartets for better filtration. We expect that at least 2-3 good quartets will have overlapping 4-tuples between the P 's and the Q 's which provides 2^{-12} filtration power. Thus we will get about $2^{32} \cdot 100 \cdot 2^{-12} \approx 2^{27}$ candidates for 64-bit partial key: 32-bits at the top and 32-bits at the bottom. The correct key will be suggested at least twice, and the wrong keys would likely be suggested only once, since we are below the birthday bound for a 64-bit event. Thus we expect that all the wrong pairs will be filtered out at the key-recovery step. Finally, complexity of this 6 round attack will be 2^{39} chosen plaintexts, 2^{71} adaptively chosen ciphertexts,

the same amount of time steps spent mainly encrypting the texts and 2^{37} bytes of memory.

It seems likely that this attack may be converted to break 7-rounds of the 192-bit key AES.

3 Conclusions

We have shown boomerang attacks on 5 and 6 round AES much faster than exhaustive search. We notice that AES has many truncated differentials with probability one spanning up to three rounds, however they are quite expensive in terms of probability when trying to extend them at either end of the boomerang distinguisher. The attacks presented in this paper are twice more efficient than the “Square” attack but are less efficient than the partial sum attack. This may mean that AES has sufficient security margin against the boomerang attacks. The attacks presented in this paper are *structural* attacks (i.e. they do not use specific properties of the underlying cipher) applicable to arbitrary 5-6 round SPNs with incomplete diffusion. It is an open problem whether the middle-round gaining trick, for example as used in a recent attack on Safer++ [1] would be applicable to the AES.

References

- [1] A. Biryukov, C. D. Cannière, and G. Dellkrantz, “Cryptanalysis of SAFER++,” in *Proceedings of Crypto’03* (D. Boneh, ed.), Lecture Notes in Computer Science, Springer-Verlag, 2003. NES/DOC/KUL/WP5/028. Full version available at <http://eprint.iacr.org/2003/109/>.
- [2] J. Daemen and V. Rijmen, *The Design of Rijndael: AES — The Advanced Encryption Standard*. Springer-Verlag, 2002.
- [3] N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, and D. Whiting, “Improved cryptanalysis of Rijndael,” in *Fast Software Encryption, FSE 2000* (B. Schneier, ed.), vol. 1978 of *Lecture Notes in Computer Science*, pp. 213–230, Springer-Verlag, 2001.
- [4] D. Wagner, “The boomerang attack,” in *Fast Software Encryption, FSE’99* (L. R. Knudsen, ed.), vol. 1636 of *Lecture Notes in Computer Science*, pp. 156–170, Springer-Verlag, 1999.