Abstract. In this paper, we report on our design of a chosen plaintext attack with work factor $2^{252}$ to recover of the first and the last subkeys of a 7-round Rijndael, while differential cryptanalysis against Rijndael have been done for up to 6 rounds and reported in published papers. We found a 5-round boomerang characteristic for Rijndael, and designed a chosen plaintext attack based on this characteristic, with work factor $2^{249}$ to recover the 32 bits of the 1st round subkey and the 32 bits of the 7th round subkey. We also designed some similar attacks to recover other bits of subkeys of the first round and the last round. Therefore the work factor of this chosen plaintext attack to recover all bits of the first and the last subkeys of a 7-round Rijndael will be $2^{252}$, that is less than exhaustive search. It means that a 7-round Rijndael will be compromised with differential boomerang cryptanalysis.

Key words: Cryptography, Block Cipher, Differential Cryptanalysis, Differential Model, Rijndael, and Ant Colony Technique.

1. Introduction

Differential cryptanalysis of block ciphers was proposed by Biham and Shamir[2]. This method of cryptanalysis is done in two phases, which we call design and execution of attack. In the design phase, a cryptanalyst finds weaknesses of a cipher algorithm and applies them to find an appropriate differential characteristic for that cipher. In the execution phase, he must gather enough ciphertext pairs with the found characteristic and then identifies the effective bits of the key according to a counting scheme. These phases can be summarized as the following steps:
I. Design of attack
1. Build the difference distribution table of S-boxes.
2. Compute the probability of all possible one-round characteristics.
3. Examine all combinations of one-round characteristics to find a suitable full-round characteristic.

II. Execution of attack
1. Gather enough ciphertext pairs with the found characteristic.
2. Identify effective bits of the key.

For an interested reader, we would like to elaborate that the bottleneck of the design phase is the step 3, i.e., examining all combinations of the one-round characteristics to find the best full-round characteristic. In [7], we described forward-backward method and applied dynamic programming and backtracking technique to used it to find suitable differential characteristics for Serpent. Although we managed to obtain suitable result, but one spend much time on it, as the examination of all combinations of one-round characteristics is a time consuming task. Then, in [8] we presented a model for finding suitable differential characteristics with applying intelligent techniques through forward-backward method. That model represents the problem of finding the best differential characteristic for a block cipher algorithm as the problem of finding the shortest path in a directed graph. Then, we applied ant-colony technique [6] for finding the shortest path in the directed graph. In this way, we reached two advantages. Firstly, by applying this technique, one can obtain a suitable result without examining the whole search space. Secondly, intelligent techniques such as ant-colony technique may reduce dependency of cryptanalysis from cryptanalyser.

Rijndael is a block cipher, which is selected as the Advanced Encryption Standard[1]. Differential cryptanalysis against Rijndael have been done for up to 6 rounds in [5], [3] and [4]. We found 8 5-round boomerang characteristics for Rijndael, and we designed a chosen plaintext attack based on these 5-round boomerang characteristics to recover of the first and the last subkeys of a 7-round Rijndael. The work factor of this attack is $2^{252}$, that is less than exhaustive search.

The remainder of this paper organized as follows: In section 2, we give a brief description of Rijndael(AES). In section 3, we review the boomerang attack. In section 4, we present our found characteristic and designed attack for a 7-round Rijndael. The last section is the conclusion of paper.

2. The description of Rijndael

Rijndael is a block cipher which is selected as the AES[1]. The AES requirements state that the length of the key can be specified to be 128, 192 or 256 bits while the length of the block is 128 bits. The designers of Rijndael cipher also allow the length of the block to be 128, 192 or 256 bits, independently of the length of the key. In this paper we deal with the variant in which the lengths of both are 256 bit. In this case the cipher consists of 14 rounds. The intermediate state is arranged in a 4×8 matrix of bytes. Every round except for the last consists of the following transformations:

1. **Byte Substitution:** This transformation is applied to each byte separately. Each byte is considered as representing coefficients of a polynomial of degree less than 8 over $\mathbb{Z}_2$ (i.e., elements of $\text{GF}(2^8)$). The inverse of this polynomial modulo $(x^8 + x^4 + x^3 + x + 1)$ is calculated, the result is multiplied by a fixed
matrix and is added to a fixed polynomial. This transformation is the only non-linear transformation in the cipher.

2. **Shift:** The first row of the matrix remains constant, the second row is shifted one byte to the right, the third row is shifted three bytes to the right, and the last row is shifted four bytes to the right.

3. **Mix Column:** Each column of the matrix is considered as a polynomial of degree less than 4 over \( \text{GF}(2^8) \) and this polynomial is multiplied by the polynomial \( 03x^3+01x^2+01x+02 \), where \( a_x \) denotes hexadecimal value modulo \( x^4 + 1 \). This transformation is linear.

4. **Add Round Key:** A 256-bit round key, which is derived from the key by the Key Expansion algorithm, is byte-wise Xored to the state.

In the last round the MixColumn transformation is omitted and before the first round an Add Round Key transformation is performed, using the key itself as a round key.

The round key of each round is derived from the key using the Key Expansion algorithm. Each round key is of length 256 bit and by the design knowing a round key of any round is enough to recover the key. Let us denote the bytes of the expanded key by \( K_0, K_1, K_2, \ldots \), where the key is \( K_0, K_1, \ldots, K_{15} \). Then the expanded key is derived from the following formula:

\[
K_n = \begin{cases} 
K_{n-1} \oplus K_{n-16} & \text{if } 16 \mid n \\
K_{n-16} \oplus \text{ByteSubstitution(Shifted } K_{n-1}) \oplus \text{Rcon} & \text{otherwise}
\end{cases}
\]

### 3. The boomerang attack

The boomerang attack is a differential attack that attempts to generate a quartet structure at an intermediate value halfway through the cipher\(^\text{[9]}\). The attack considers four plaintexts \( I_1, I_2, I_1', I_2' \) along with their respective ciphertexts \( O_1, O_2, O_1', O_2' \). Let \( E(.) \) represents 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 for \( E_0 \), say \( \Delta \rightarrow \Delta^* \), as well as a differential characteristic for \( E_1^{-1} \) as \( \nabla \rightarrow \nabla^* \).

We want to cover the pair \( (I_1, I_2) \) with the characteristic \( \Delta \rightarrow \Delta^* \) for \( E_0 \), and to cover the pairs \( (I_1, I_1') \) and \( (I_2, I_2') \) with the characteristic \( \nabla \rightarrow \nabla^* \) for \( E_1^{-1} \). Then the pair \( (I_1', I_2') \) is perfectly set up to use the characteristic \( \Delta \rightarrow \Delta^* \) for the inverse of the first half of the cipher. When this characteristic also holds, we will have the same difference in the plaintexts \( I_1' \) and \( I_2' \) as found in the original plaintexts \( I_1 \) and \( I_2 \). This is why it is called the boomerang attack: when you send it properly, it always comes back to you.
The same attack works even if we do not predict the exact value of $\nabla^*$ ahead of time, but instead merely require that the difference after decrypting by $E_1$ is the same in the two pairs $(I_1,I'_1)$ and $(I_2,I'_2)$. A similar observation also holds for $\Delta^*$. Therefore, we may sum over all values for $\Delta^*$ and $\nabla^*$ to obtain the probability of boomerang characteristic as follows:

$$
Pr \approx \sum_{\Delta^*} \Pr(\Delta \rightarrow \Delta^* \text{ by } E_0)^2 \times \sum_{\nabla^*} \Pr(\nabla \rightarrow \nabla^* \text{ by } E_1^{-1})^2
$$

4. Design the boomerang attack against 7-round Rijndael

Differential cryptanalysis against Rijndael have been done for up to 6 rounds and is reported in [5], [3], and [4]. We found a two-round differential characteristic with probability of $2^{-30}$, as shown in table 1, and a three-round differential characteristic with probability of $2^{-126}$, as shown in table 2. Each row of these tables contains three columns. The first column gives the round number, the second column gives the input difference or the output difference of the round substitution transformation, and the last column gives the probability of the one-round differential characteristic. Note that the input difference of each round is obtained by applying the shift and mix-column transformations to the output difference of the previous round.

We designed a boomerang attack based on these characteristics for five round of Rijndael. If we call the first two rounds of this five rounds $E_0$ and the final three rounds $E_1$, the five-round Rijndael is $E = E_0 \circ E_1$. Let

<table>
<thead>
<tr>
<th>Round #</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Input \ Output</td>
<td>$2^6$</td>
</tr>
<tr>
<td>3</td>
<td>Input \ Output</td>
<td>$2^{24}$</td>
</tr>
</tbody>
</table>
### Table 2: Three-round differential Characteristic for Rijndael

<table>
<thead>
<tr>
<th>Round #</th>
<th>Input</th>
<th>Output</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>A80000E7F2CB000000080000A000E4000AB0CE7A6006800BE0000500000055CC</td>
<td>A10000F7E7C000000530000870000A0001F7F74702600CD0000300000009F28</td>
<td>$2^{-96}$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>9000000000000000000000000000000000000000000000000000000000000000</td>
<td>0300000000000000000000000000000000000000000000000000000000000000</td>
<td>$2^{-24}$</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>3400000000000000000000000000000000000000000000000000000000000000</td>
<td>7800000000000000000000000000000000000000000000000000000000000000</td>
<td>$2^{-6}$</td>
<td></td>
</tr>
</tbody>
</table>
| 7       | F67B7B8D000000000000000000000000000000000000000000000000000000000 | Input2→Output3 as our characteristic for E0 and Output6→Input4 as our differential characteristic for E1⁻¹. The probability of this five-round boomerang characteristic is $\left(2^{-30}\right)^2 \times \left(2^{-128}\right)^2 = 2^{-312}$. As we mentioned earlier, we can design a similar characteristic even if we do not predict the exact values of Output3 and Input4. Therefore we may sum over all values for Output3 and Input4 in two-round characteristics of Input2→Output3 and in three-round characteristics of Output6→Input4 respectively. The number of active Sboxes in these characteristics is 5 and 21 respectively. According to the differential distribution table of the Sbox of Rijndael, the probability of each active Sbox is $2^{-6}$ and $2^{-7}$ in 1 and 127 cases respectively. The probability of five-round boomerang characteristic is calculated as follows:

$$\Pr = \sum_{i=0}^{5} \left(2^{-34.83} \times 2^{-146.30}\right)^{2} \cong 2^{-182}$$

We extended above five-round boomerang characteristic to a seven-round attack to recover the 32 bits of the 1st round subkey and the 32 bits of the 7th round subkey corresponding to the 4 active Sboxes of Input2 and Input7 respectively. This attack can be done in following steps (figure 2):

1- Set a counter corresponding to each of the probable values of the 32 bits of the 1st round subkey and the 32 bits of the 7th round subkey of Rijndael and initialize it by zero.

2- Gather enough pairs with difference of Input2, denote such a pair as ($I1,I2$). Do step 3 to step 10 for each pair.

3- Apply the inverse of the first round for each of the pairs by employing all probable values of the 32 bits of 1st round subkey to determine the bits of the pairs that are corresponding to the active sboxes, and determine the other bits of the pairs randomly. Denote such a resulted pair as ($P1,P2$).

4- Apply 7-round Rijndael cipher to each of plaintext pairs to obtain ciphertext pairs, Denote such a ciphertext pair as ($C1,C2$).

5- Peel off the last round for each of the ciphertext pairs corresponding to each of the probable values of the 128-bit subkey, Denote such a resulted pair as ($O1,O2$).

6- Compute $O'i=Oi\oplus\nabla$ for $i=1,2$, where $\nabla=Input_{7}$.

7- Re-encrypt the last round with the guessed 32-bit last round subkey, Denote such a resulted pair as ($C'1,C'2$).

8- Apply 7-round Rijndael decipher to resulted pair, Denote such a resulted pair as ($I'1,I'2$).

9- Re-encrypt the first round with the guessed 32-bit first round subkey to obtain boomerang pairs, Denote such a boomerang pair as ($Q1,Q2$).
10- If difference of boomerang pair is equal $\text{Input}_2$ increase the counter corresponding to guessed subkeys.

11- The subkey corresponding to counter with maximum value is the right subkey.

The bold edges show the steps of the attack as above algorithm, where the label of each edge gives the step number.

We can identify all but approximately $2^{228}$ of our boomerang pairs are wrong pairs, because their differences of $(I'_1, I'_2)$ cannot correspond to our desired difference $\text{Input}_2$. The signal to noise is calculated as follows:

$$S/N = \text{Pr} \times 2^K / \alpha \beta = 2^{-182} \times 2^{64} / 1 \times 2^{-228} = 2^{110}$$

Therefore only 3 or 4 right pairs are required, then we need $4 \times 2^{182} = 2^{184}$ pairs with our desired difference $\text{Input}_2$ in step 1 of above algorithm. The work factor is calculated as follows:

$$WF = 2 \times 2^{184} \times 2^{64} = 2^{249}$$

We found 7 other 5-round boomerang characteristics with probability of $2^{-182}$ as shown in Appendix 1. We designed 7 attacks based on these characteristics similar to the above attack, such that each recover 32 bits of the first subkey and 32 bits of the last subkey in a 7-round Rijndael, while the work factor of each attack is $2^{249}$. Therefor the work factor of attack to recover the first and the last subkeys of a 7-round Rijndael is $2^{252}$.

5. Conclusion

In this paper we found a 5-round boomerang characteristic, and designed a chosen plaintext attack based on this 5-round boomerang characteristic, with work factor of $2^{249}$ to recover 32 bits of the first round subkey and 32 bits of the last round subkey of a 7-round Rijndael. Then we found 7 other 5-round boomerang characteristics with probability of $2^{-182}$ that one can design 7 similar attacks to recover other bits of these Rijndael.
subkeys. The work factor of this attack will be $2^{252}$, that is less than exhaustive search. It means that 7-round Rijndael will be compromised with differential boomerang cryptanalysis.

References


Appendix 1: Seven 5-round boomerang differential characteristics for Rijndael.

We found seven 5-round boomerang differential characteristics, as shown in the following tables. Each row of these tables contains three columns. The first column gives the round number, the second column gives the input difference or the output difference of the round substitution transformation, and the last column gives the probability of the one-round differential characteristic.

<table>
<thead>
<tr>
<th>Round #</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output</td>
<td>$2^{0}$</td>
</tr>
<tr>
<td>2</td>
<td>Input Output</td>
<td>$2^{6}$</td>
</tr>
<tr>
<td>3</td>
<td>Input Output</td>
<td>$2^{24}$</td>
</tr>
<tr>
<td>4</td>
<td>Input Output</td>
<td>$2^{96}$</td>
</tr>
<tr>
<td>5</td>
<td>Input Output</td>
<td>$2^{24}$</td>
</tr>
<tr>
<td>6</td>
<td>Input Output</td>
<td>$2^{6}$</td>
</tr>
<tr>
<td>7</td>
<td>Input</td>
<td>$2^{6}$</td>
</tr>
</tbody>
</table>
2. Round # | Differential Characteristic | Probability
--- | --- | ---
1 | Output | 0000000000000000072000000004B00000000000000000200000000E2000000000
2 | Input | 0000000000000000000560000000000000000000000000000000000000000000
Output | 2^{-6}
3 | Input | 0000000000000000000AC5656FA000000000000000000000000000000000000000
Output | 2^{-24}
4 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
5 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-24}
6 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
7 | Input | 0000000000000000000000000000000000000000000000000000000000000000

3. Round # | Differential Characteristic | Probability
--- | --- | ---
1 | Output | 0000000000000000000720000000000000000000000000000000000000000000
2 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
3 | Input | 0000000000000000000AC5656FA000000000000000000000000000000000000000
Output | 2^{-24}
4 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
5 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-24}
6 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
7 | Input | 0000000000000000000000000000000000000000000000000000000000000000

4. Round # | Differential Characteristic | Probability
--- | --- | ---
1 | Output | 0000000000000000000000000000000000000000000000000000000000000000
2 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
3 | Input | 0000000000000000000AC5656FA000000000000000000000000000000000000000
Output | 2^{-24}
4 | Input | ABACE7A6006800E0000050000000055CCA80000E7F2CB000000080000AE00E4000
Output | 2^{-6}
5 | Input | 01F2F74702600CD00030000009F28A10000F7EA7C000053000870001A001F2F747
Output | 2^{-24}
6 | Input | 0000000000000000000000000000000000000000000000000000000000000000
Output | 2^{-6}
7 | Input | 0000000000000000000000000000000000000000000000000000000000000000
<table>
<thead>
<tr>
<th>Round #</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Input: AE00E400ABACE7A6006800BE0000500000055CCA80000E7F2CB00000080000</td>
<td>$2^{-6}$</td>
</tr>
<tr>
<td>5</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Input:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Round #</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Input: F2CB000000E00000E00E400ABACE7A6006800BE0000500000055CCA80000E7F2CB00000080000EA7C00000S30000870000A0001F2F747002600CD000030000009F28A10000F7EA7C00000530000</td>
<td>$2^{-6}$</td>
</tr>
<tr>
<td>5</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Input:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Round #</th>
<th>Differential Characteristic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Input:</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Output</td>
<td></td>
</tr>
</tbody>
</table>