

New Impossible Differential Attacks on AES

AES

➤ 16 byte data, 128/192/256 bits key.

➤ All operations in AES are byte-based.

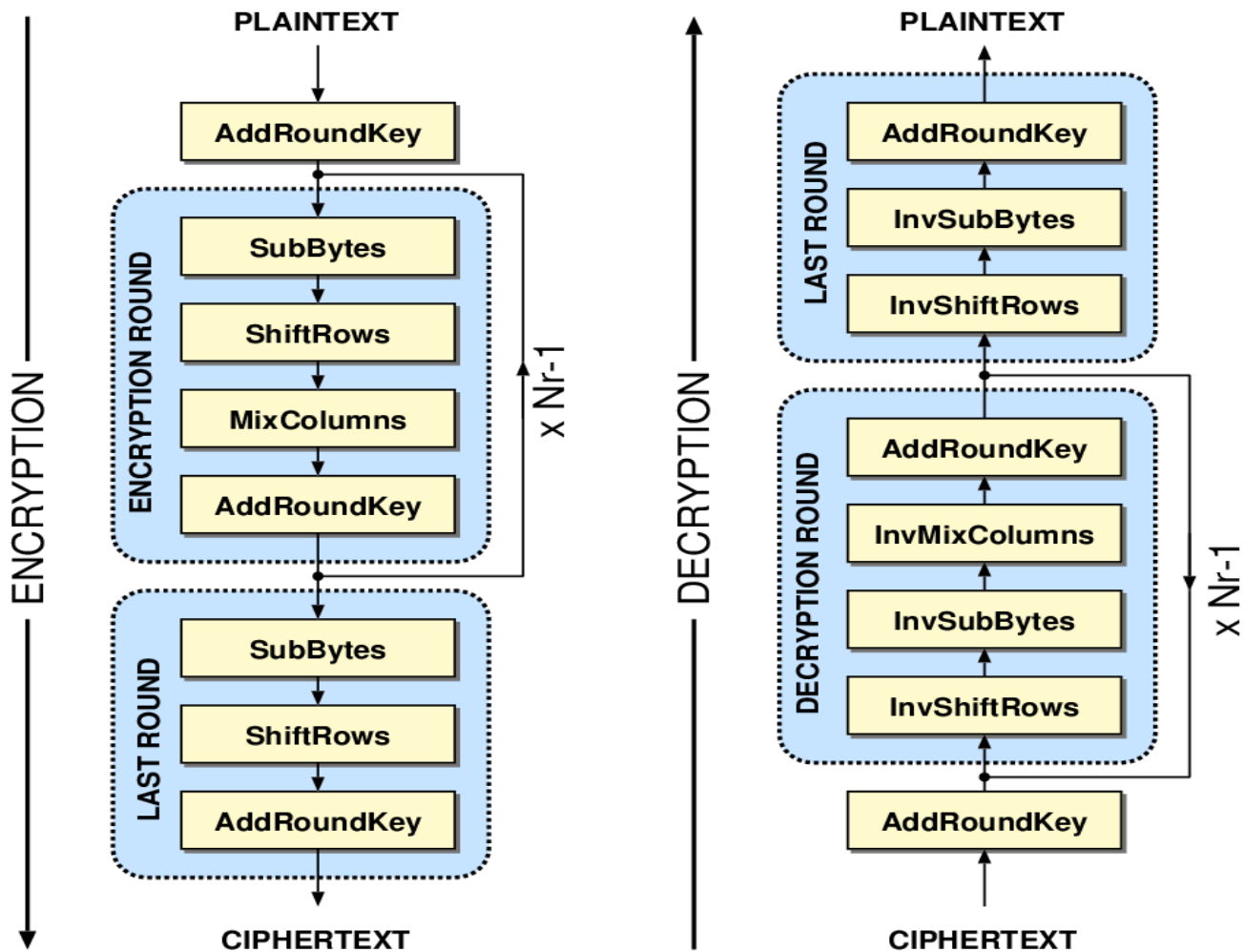
The state consists of 128 bits = 16 bytes, viewed as a 4x4 array of bytes.

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

AES

- Four Operations:
 - SubBytes (SB).
 - ShiftRows (SR).
 - MixColumns (MC).
 - AddRoundKey (ARK).

AES Rounds



Notations

- $x_i^I, x_i^{SB}, x_i^{SR}, x_i^{MC}, x_i^O.$
- $x_{i,Col(z)}.$
- $k_i.$
- $SR(Col(i)), SR^{-1}(Col(i)).$
- $w_i=MC^{-1}(k_i).$

Impossible differential

- Proposition 1:

If:

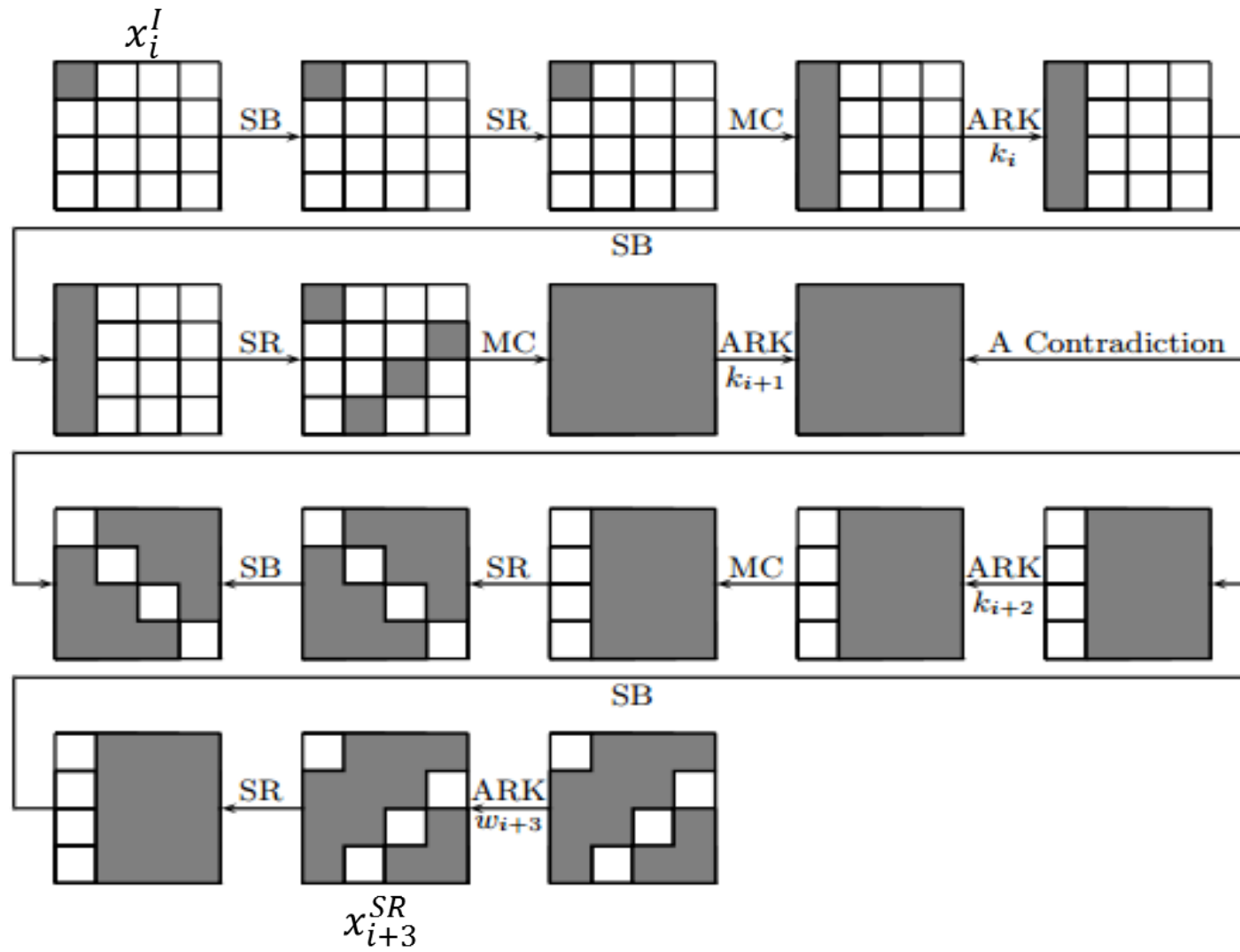
- $\Delta(x_i^I)$ has only one non-zero byte.
- in $\Delta(x_{i+3}^{SB})$, at least one of the four sets $SR(\text{Col}(i))$ is equal to zero.

0	1	2	3
1	2	3	0
2	3	0	1
3	0	1	2



Then:

$\Delta(x_i^I) \rightarrow \Delta(x_{i+3}^{SR})$ is impossible differential.

Proof:



Bahrak-Aref Attack on 7-round AES-128

1. Encrypt $2^{85.5}$ structures of 2^{32} , such that bytes $SR^{-1}(Col(0))$  are all the 2^{32} possible values, the rest bytes are fixed.
2. Discard all cipher text pairs with non-zero difference in bytes $SR(Col(1,2))$, keep .
3. Guess the values of $k_{6,SR(Col(3))}$ and partially decrypt the cipher text pairs through round 6 to get the difference $\Delta(x_{5,Col(3)}^{SR})$, select only pairs with $\Delta(x_{5,Col(3)}^{SR})$ has a non-zero in byte 7.

4. Guess the values of $k_{6,SR(Col(0))}$ and partially decrypt the cipher text pairs through round 6 to get the difference $\Delta(x_{5,Col(0)}^{SR})$, select only pairs with $\Delta(x_{5,Col(0)}^{SR})$ has a non-zero in byte 0.
5. Guess the values of bytes (0,7) of w_5 and partially decrypt the cipher text pairs through round 5 to get the difference $\Delta(x_{4,Col(0)}^{SR})$, select only pairs with $\Delta(x_{4,Col(0)}^{SR})$ has one zero byte value.
6. For each of the remaining pairs, consider the corresponding pair and discard all the values of $k_{-1,SR(Col(0))}$ that lead to the Input difference of the impossible difference in the input of round 1.
 - If a guess for these bytes remains, guess all the remaining key bytes, otherwise repeat steps 4-6 for a diff guess of k_6 .

BA attack Analysis

- Time = 2^{121} 7-round AES encryption
- Data = $2^{117.5}$ chosen plaintext.
- Memory = 2^{109} bytes of memory (required for storing discarded values).
- Ideas for improvements:
 - Instead of partially decryptions each pair under many possible values key guesses, we can use table look ups to deduce which key it suggest.
 - Re-using the data and repeating several times a slightly different impossible differential.

Improving BA's time complexity

- Step 1,2 remain unchanged except reduction in the number of plaintexts.
- Observation 1: Given an input and an output differences of the SubBytes operation, there's on average one pair of actual values that satisfies these differences.
- In step 3, instead of guessing the 32 bits of k_6 , we use observation 1 to improve it. We know $\Delta(x_{6,Col(3)}^{SB})$, we will show how to find $\Delta(x_{6,Col(3)}^I)$.
 - Note that there are $2^8 - 1$ (255) values for $\Delta(x_{5,Col(3)}^{SR})$ in which only byte 7 is non-zero.

Improving BA's time complexity

- Step 3:
 - Initialize 2^{32} empty lists, each corresponds to a different guess of $k_{6,SR(Col(0))}$.
 - For each remaining cipher text, for each one of the 255 differences in $\Delta(x_{6,Col(3)}^I)$, compute the key which leads this specific pair to this specific difference. Add this pair to the list corresponding to that specific key guess.
- We expect one key suggested on average. These $2^{79.2} \times 255 = 2^{87.2}$ suggestions are distributed over 2^{32} possible subkeys, and thus, for a given subkey guess, we expect $2^{55.2}$ pairs to remain.
- Time complexity: $2^{87.2}$ memory accesses.

Improving BA's time complexity

- Step 4 improvement the same as step 3.
 - As a result the time complexity is $2^{85.2}$ memory accesses.
 - The number of remaining pairs is $2^{31.2}$.
- Step 5: we can use observation 1 to improve this step.
 - We want to find bytes (0,7) of w_5 using observation 1. we already know $\Delta(x_{5,Col(0)}^{SB})$, and we want to find $\Delta(x_{5,Col(0)}^I)$.
 - There are $4 \cdot 255^3$ values $u = \Delta(x_{4,Col(0)}^{SR})$ has one zero byte value.
 - For each u there's a unique $v = \Delta(x_{5,Col(0)}^I)$ such that $v = MC(u)$.
 - Of the $4 \cdot 255^3$ possible v 's there are $4 \cdot 255 \approx 2^{10}$ with zero difference in bytes (8,12).
 - the goal is to check for every guess of bytes (0,7) of w_5 whether the value in bytes (0,4) of $\Delta(x_{5,Col(0)}^I)$ falls into these 2^{10} values.

Improving BA's time complexity

- At the end of step 5, the $2^{31.2}$ pairs and the 2^{10} possible differences propose $2^{41.2}$ candidate keys which are scattered among 2^{16} candidate values.
 - $2^{25.2}$ pairs remains.
 - $2^{64} \cdot 2^{31.2} \cdot 2^{10} = 2^{105.2}$ memory accesses.

Improving Step 6 of the BA attack & reducing Data Complexity

- Only $2^{25.2}$ pairs are analyzed in Step 6, this number is far from being sufficient to discard all the possible values of $k_{-1,SR^{-1}(Col(0))}$.
- Observation 2: the attacker uses only pairs that have a non-zero difference only in byte 0.
 - Actually there are 4 impossible differentials in this column.
 - Furthermore, for each guess of $k_{6,SR(Col(0,3))}$, we apply for similar attacks: The first is the BA attack, and other three variants, bytes (0,7) of w_5 are replaced by the pairs of bytes (4,11), (8,15) and (12,3). Each attack discards possible values of 112 subkey bits ($k_{6,SR(Col(0,3))}$, $k_{-1,SR^{-1}(Col(0))}$ and 16 bits differ between the attacks).
 - We can run modified steps 3-5 several times.

Improved BA Complexity

	Before	After
Data Complexity	$2^{117.5}$	$2^{112.2}$
Time Complexity	2^{121} encryptions	$2^{117.2}$ memory accesses
Memory Complexity	2^{109}	$2^{93.2}$

Backup

Extension BA to 8-round AES-256

- The obvious extension is guessing the 8th round subkey and applying the 7-round attack.
 - Time complexity: $2^{237.2}$ encryptions & $2^{245.7}$ memory accesses.
- We improve this basic attack by exploiting the key schedule and modifying the way of partial decryption of round 7 is handled.
 - We note that the knowledge of k_7 , leads the knowledge of $k_{5,Col(1,2,3)}$. We will slightly change the attack to depend on the bytes of $w_{5,Col(1,2,3)}$, this way we eliminate the need of “guess” bytes of w_5 .
 - We can handle the partial decryption in amore effeceind way, we guess 96 bits of k_7 , then try all the possible 2^{16} differences

8-round AES-256 Attack Algorithm

1. Guess bytes of $k_{7,SR(Col(0,1,2))}$ and decrypt all the cipher texts through round 7 in three columns to get $\Delta(x_{6,Col(0,1,2)}^{SR})$, discard all the pairs in which $\Delta(x_{6,Col(0,1,2)}^{SR})$ is not equal to zero in at least of the bytes (0,1,4,10,12,13).
2. Initialize for each guess of $k_{7,Col(3)}$ an empty list.
3. For each of the remaining pairs, and each of the possible differences in $\Delta(x_{6,Col(3)}^{SR})$ in which bytes (7,11) are zero, and find the key that this pairs suggests (using observation 1). And add the pair to the list corresponding to this key.

8-round AES-256 Attack Algorithm

4. Repeat the 7-round attack with the following changes:
 - a) In Step 3 of 7-round attack, the attacker guesses $w_{6,SR(Col(3))}$ and selects only pairs in which $\Delta(x_{5,Col(3)}^{SR})$ has a non-zero value only in byte 3.
 - b) In Step 4 of 7-round attack, the attacker guesses $w_{6,SR(Col(2))}$ and selects only pairs in which $\Delta(x_{5,Col(2)}^{SR})$ has a non-zero value only in byte 6.
 - c) The attacker then partially decrypts the pair (using the knowledge of the relevant bytes of w_5), and checks whether the obtained difference is active only in three bytes of x_4^{MC} .