

Impossible Differential Key Recovery Attack on Five-round AES

Impossible Differential Key Recovery Attack (Five-round AES)

1. Pair Collection

- Attacker uses **chosen-plaintext model** with difference ΔP .
- Probability a random pair satisfies output difference ΔC is 2^{-32} .
- From 2^N queries, expected pairs with $\Delta C = 2^{N-32}$.

2. Recovery of Subkey $sk_0[0, 5, 10, 15]$

- Focus on 4 bytes of the first round key.
- Since AddRoundKey doesn't change differences, $\Delta P = \Delta S_{1I}$ (known).
- Attacker picks **255 possible values of ΔS_{1MC}** (nonzero byte differences).
- Each maps to ΔS_{1SB} via inverse AES operations, stored in table $T[i]$.
- Using AES S-box properties, pairs produce expected **1 solution per trial**.
- Each pair yields **255 wrong subkey candidates**, which can be discarded.

3. Number of Pairs Needed

- Each useful pair removes **255 keys**, so total wrong key eliminations $\approx 2^{N-24}$.
- Goal: narrow down 2^{32} subkey space to the single correct subkey.
- Simply reaching 2^{32} eliminations ($N = 56$) is not sufficient since duplicates exist.
- Must repeat elimination until **key space shrinks to 1** (using memory κ of size 2^{32} to track candidates).

Key Idea: The attacker gradually removes impossible subkeys by exploiting the impossible differential property, reducing the key space until only the correct subkey remains.

Key Space Reduction & Complexity (Five-round AES Attack)

1. Key Space Reduction

- Initial subkey space = 2^{32} .
- One wrong key suggestion reduces space by factor:

$$\left(1 - \frac{1}{2^{32}}\right)$$

instead of subtracting 1 (to handle duplicates).

- After r wrong key suggestions (manipulating the value of r):

$$2^{32} \cdot \left(1 - \frac{1}{2^{32}}\right)^r \approx 2^{32} \cdot e^{-r/2^{32}}$$

- With 2^{32} wrong suggestions, space $\approx 2^{30.56}$ (still too large).
- With $2^{32} \cdot 16$ wrong suggestions, space $\approx 2^{8.92}$.
- With $2^{32} \cdot 32$ wrong suggestions, space $< 1 \rightarrow$ unique key identified.
- Requires about 2^{37} wrong key suggestions.

2. Number of Plaintext Pairs

- From pair collection: $\approx 2^{N-24}$ wrong key suggestions.
- Need 2^{37} , so:

$$2^{N-24} \geq 2^{37} \implies N = 61$$

- Thus, need 2^{61} plaintext pairs.

3. Attack Procedure (Algorithm 4.7)

- Collect 2^{61} plaintext pairs.
- For each pair, derive wrong key candidates and discard them.
- Iterate until key space shrinks to size 1 \rightarrow recover $sk_0[0, 5, 10, 15]$.

4. Complexity Evaluation

- **Data:** 2^{62} chosen plaintexts.
- **Time:** $\sim 2^{61}$ XOR operations (less than 2^{61} AES-5 computations).
- **Memory:**

– 2^{32} entries for κ (subkey space),

- 2^8 for $T[i]$ table,
- 2^{29} plaintext pairs.

- **Overall Complexity:**

$$(\text{Data, Time, Memory}) = (2^{62}, 2^{61}, 2^{32})$$

Key Point: By iteratively discarding impossible subkeys with 2^{61} plaintext pairs, the attacker can recover the 32-bit subkey $sk_0[0, 5, 10, 15]$ with feasible time and memory.