Linear Cryptanalysis: From Intuition to Full Examples

Masks, Correlations, and Matsui's Attack

Generated for You

January 12, 2026

1 Introduction: The Goal

Linear Cryptanalysis (LC) is a **Known-Plaintext Attack**: you passively observe plaintext–ciphertext pairs and look for statistical patterns that should not exist in a random permutation.

The aim is to find linear relations that slightly bias certain bit parities, then exploit these biases to distinguish the cipher from random and recover key information.

2 Core Concept: Correlation

The key idea is correlation between input and output bit parities.

- **Correlation +1:** Perfect copy. If $A = 1$, then $B = 1$; if $A = 0$, then $B = 0$.
- **Correlation -1:** Perfect opposite. If $A = 1$, then $B = 0$ and vice versa; this is still a perfect dependency.
- **Correlation 0:** Statistical independence. Knowing A gives no information about B.

Common Mistake: Correlation vs. Independence

"1 always leads to 0" is strong *anti-correlation*, not independence. Independence means that given A, B is 0 or 1 with probability 50% each.

3 Linear Masks and Parities

3.1 Dot Product with a Mask

LC does not use raw bits but linear expressions built with masks α, β, γ over plaintext P, ciphertext C, and key K:

$$(P \cdot \alpha) \oplus (C \cdot \beta) = (K \cdot \gamma).$$

Intuition: The Bitwise Dot Product

For a bitstring P and mask α:

1. Select the positions where α has 1s (bitwise AND conceptually).
2. XOR all selected bits.
3. The result is a single parity bit in $\{0, 1\}$.

The mask acts as a linear "compressor" from many bits to one parity bit.

Example: Bitwise Masking Step-by-Step

Compute $P \cdot \alpha$:

- Input $P = 10110$
- Mask $\alpha = 00110$ (inspect 3rd and 4th bits)

Selection (conceptual AND):

$$P = 1\ 0\ \mathbf{1}\ \mathbf{1}\ 0, \quad \alpha = 0\ 0\ \mathbf{1}\ \mathbf{1}\ 0$$

Selected bits: $1, 1$.
Compression (XOR parity):

$$0 \oplus 0 \oplus \mathbf{1} \oplus \mathbf{1} \oplus 0 = \mathbf{0}.$$

So $P \cdot \alpha = 0$ for this example.

4 Correlation Intuition

4.1 Copy, Rebel, Stranger

Consider an input bit A and output bit B.

Scenario A: Copycat $(+1)$ Always $B = A$. Perfect predictability.

Scenario B: Rebel (-1) Always $B = \overline{A}$. Still perfectly predictable: just flip the bit.

Scenario C: Stranger (0) Half the time $B = A$, half the time $B \neq A$; A is useless for predicting B.

In LC, correlations close to ± 1 are extremely exploitable; correlations near 0 are useless.

5 Linear Approximation Table (LAT)

For an S-box S, the Linear Approximation Table captures how input and output parities are correlated.

- Rows: input masks α.
- Columns: output masks β.
- Entry: bias or correlation of $(\alpha \cdot x)$ and $(\beta \cdot S(x))$ over all inputs x.

5.1 Example: One LAT Entry

Consider a 3-bit S-box S. Suppose we test:

- Input mask $\alpha = 101$ giving $x_1 \oplus x_3$.
- Output mask $\beta = 010$ giving y_2.

We check whether $x_1 \oplus x_3 = y_2$ over all 8 inputs.

Input x			Output $y = S(x)$			Masked In	Masked Out	Match?
x_1	x_2	x_3	y_1	y_2	y_3	$x_1 \oplus x_3$	y_2	
0	0	0	1	**1**	0	0	1	No
0	0	1	0	**0**	1	1	0	No
0	1	0	0	**0**	0	0	0	**Yes**
0	1	1	1	**1**	1	1	1	**Yes**
1	0	0	1	**0**	0	1	0	No
1	0	1	0	**1**	0	0	1	No
1	1	0	0	**1**	0	1	1	**Yes**
1	1	1	0	**0**	1	0	0	**Yes**

Matches: 4 out of 8, so the probability is $p = 1/2$ and the bias $\epsilon = p - 1/2 = 0$.

Intuition: Interpreting Bias

Bias 0 means the chosen masks are useless for LC on this S-box: the masked input and masked output behave independently. Useful approximations are those where matches are very frequent (e.g., 7/8) or very rare (e.g., 1/8).

6 Piling-Up Lemma and Linear Trails

Real ciphers have many rounds. One constructs a *linear trail* by chaining S-box approximations across rounds.

If the correlation of round i along the trail is corr_i, then under independence assumptions:

$$\text{Total Correlation} = \prod_i \mathrm{corr}_i.$$

Each correlation has magnitude less than 1, so the overall bias shrinks exponentially with the number of rounds, which is why adding rounds greatly improves resistance to LC.

7 Matsui's Algorithms

7.1 Algorithm 1: Sign Test for One Key Bit

This algorithm recovers the single bit value of a key parity $(K \cdot \gamma)$.

1. For N known plaintext–ciphertext pairs, evaluate $(P \cdot \alpha) \oplus (C \cdot \beta)$.
2. Count how many times the result is 0; call this T_0.
3. If $T_0 > N/2$, guess key bit 0; if $T_0 < N/2$, guess key bit 1.

7.2 Algorithm 2: Last-Round Subkey Recovery

1. Build a linear trail covering rounds 1 to $R - 1$.
2. For each candidate last-round subkey k', partially decrypt one round.
3. For each k', measure the correlation predicted by the trail.
4. The key k' with the largest absolute correlation is taken as the correct subkey.

Intuition: Signal vs. Noise Key Guesses

Wrong subkeys produce random-looking intermediate values, so measured correlations are near 0. The correct subkey "aligns" the trail, producing a clear correlation peak close to the theoretical value from the Piling-Up Lemma.

8 Recap

1. Linear masks select bits and XOR them to form parities.
2. Correlation measures whether those parities depend on each other; ± 1 is perfect, 0 is independence.
3. LAT entries quantify biases for S-box mask pairs; strong biases are attack targets.
4. The Piling-Up Lemma combines per-round correlations into a trail across many rounds.
5. Matsui's algorithms use biased linear relations to recover key bits and last-round subkeys from known plaintexts.