3 Cryptanalysis of Double Ciphers

Meet in the Middle

The name of this attack against double encryption goes back to Merkle and Hellman in 1981. (Don't confuse it with the "Man in the Middle" attack against cryptographic protocols.) They formalized an attack that worked in "classical times" against rotor machines, see the web page `http://www.staff.uni-mainz.de/pommeren/Cryptology/Classic/5_Rotor/AnalRot.html`.

Consider the composition of two encryption functions with different keys:

$$\begin{array}{ccccc} \Sigma^* & \xrightarrow{f_k} & \Sigma^* & \xrightarrow{f_h} & \Sigma^* \\ a & \mapsto & b & \mapsto & c. \end{array}$$

Assume a pair (a, c) of corresponding plaintext and ciphertext is known, and assume that the exhaustion of the simple cipher is feasible. Then the attacker builds two tables:

- all $f_k(a)$, $k \in K$,
- all $f_h^{-1}(c)$, $h \in K$,

and compares them. Each coincidence yields a possible pair $(h, k) \in K^2$ of keys that can be further inspected, say with further known (or guessed) plaintext.

Expenses

This attack needs

- $2 \cdot \#K$ encryptions (*not* $(\#K)^2$),
- $2 \cdot \#K$ memory cells.

Noting that we need only store one of the two tables we even halve the number of memory cells.

With the usual prefixes for memory sizes

2^{10}	2^{20}	2^{30}	2^{40}	2^{50}	2^{60}
Kilo	Mega	Giga	Tera	Peta	Exa

and using 1 byte = 8 bits we see that 60 bit keys need memory that exceeds the (actually) available capacities. However for cryptanalysis the time requirements are more critical than memory requirements. Therefore as a general finding we may state:

> *The security of a double cipher is not significantly better than the security of the underlying simple cipher.* In particular the bitlength of a key exhaustion is not doubled but only increased by 1 bit.

False Alarms

One question yet awaits an answer: How many of the coincidences in comparing the two tables lead to a wrong pair of suspected keys? That is, how likely are false alarms?

Here is a heuristic consideration: Assume we encrypt n-bit blocks with l-bit keys. Then the tables have 2^l entries, resulting in 2^{2l} comparisions. Since the number of possible values is 2^n we expect about $N_1 = 2^{2l-n}$ coincidences. (Implicitly assuming that the values behave like random. By the Birthday Paradox we expect the first coincidence after $2^{n/2}$ trials, but this is irrelevant in the present context.)

If we test the pitched key pairs with a second known plaintext block, then we are left with $N_2 = N_1/2^n = 2^{2l-2n}$ candidates. After testing t known plaintext blocks we expect to keep $N_t = 2^{2l-tn}$ candidates—but of course at least one, the right one.

Thus in general we find a unique solution as soon as

$$t \geq \frac{2l}{n}.$$

Examples

1. DES, $n = 64$, $l = 56$: $N_1 = 2^{48}$, $N_2 = 2^{-16}$. *We need about* 2 *blocks of known plaintext.*

2. IDEA, $n = 64$, $l = 128$: $N_1 = 2^{192}$, $N_2 = 2^{128}$, $N_3 = 2^{64}$, $N_4 = 1$. *We need about* 4 *blocks.*

3. AES, $n = 128$, $l = 128$: $N_1 = 2^{128}$, $N_2 = 1$. *We need about* 2 *blocks.* But the number $\#K = 2^{128}$ will by far exceed our time and memory resources (as in Example 2).

Time-Memory-Tradeoff

A more general consideration yields a "Time Memory Tradeoff": Undertaking a Meet in the Middle attack we may spare memory, allowing more execution time, by generating only partial tables:

If during a pass we fix s bits of both h and k, then we need 2^{l-s} memory cells for both of the tables of $f_k(a)$'s and $f_h^{-1}(c)$'s. As a compensation we have to go through 2^{2s} passes. The expenses are:

$2 \cdot 2^{l-s}$	encryptions for building one pair of tables,
2^{2s}	comparisions of one pair of tables, in total
$2 \cdot 2^{l+s}$	encryptions,
$2 \cdot 2^{l-s}$	memory cells.

Multiplying the number of encryptions and the number of needed memory cells we get $4 \cdot 2^{2l}$, independently from s. *This gives the attacker some freedom in using her resources in a flexible way.*

Example DES: If the attacker owns 128 terabytes of memory, she can generate 2 tables of 2^{40} blocks each, hence choose $s = 56 - 40 = 16$. Then she needs $2 \cdot 2^{72}$ encryptions. This is feasible, at least for the world's largest secret service.

Summary

Double ciphers don't improve the security of encryption in a worthwile way.