HWSec: exam

R. Pacalet

2024-04-23

You can use any document but communicating devices are strictly forbidden. Please number the different pages of your paper and indicate on each page your first and last names. You can write your answers in French or in English, as you wish. Precede your answers with the question's number. If some information or hypotheses are missing to answer a question, add them. If you consider a question as absurd and thus decide to not answer, explain why. If you do not have time to answer a question but know how to, briefly explain your ideas. Note: copying verbatim the slides of the lectures or any other provided material is not considered as a valid answer. Advice: quickly go through the document and answer the easy parts first.

The first question is worth 4 points. The 3 other questions are worth 2 points each. The problem is worth 10 points.

1 Questions

1.1 Errors during side channel attacks

During the lecture and the labs about side-channel attacks we saw that most of them retrieve a secret piece by piece (e.g., one bit at a time, 6 bits at a time...). We saw also that the attacks sometimes make errors and wrongly guesses some of these pieces.

- Give an example of side-channel attack (target, attack principle) for which such errors, even if not fixed, do not prevent the retrieval of subsequent pieces of the secret.
- Give an example of side-channel attack (target, attack principle) for which such errors, if not fixed, prevent the retrieval of subsequent pieces of the secret.
- In the second case explain how an attacker can detect that an error has been done and that the attack does not work any more.
- In the second case explain how an attacker can recover from such errors and retrieve the secret anyway.

1.2 Blinding

Blinding is a countermeasure against some of the attacks we studied during the lectures and labs. What is it? Against what kind of attacks can it be used? Give an example of a cryptographic algorithm for which blinding is not

possible and explain why. Give an example of a cryptographic algorithm and an attack for which blinding is possible and propose a concrete blinding-based countermeasure. Discuss the efficiency, the cost, the advantages and drawbacks of your countermeasure.

1.3 On-board probing attacks

Explain why Markus Kuhn's attack against the DS5002FP worked. Enumerate the characteristics of his attack scenario that contributed his success.

1.4 Power attacks

We consider a simple straightforward RSA software implementation (no Chinese Remainder Theorem, no blinding) similar to the one seen in course. With public modulus n, w-bits private exponent d and input message m, it computes signature $m^d \mod n$ using a square and multiply modular exponentiation, one exponent bit per iteration:

1: $a \leftarrow 1$
2: **for** $k \leftarrow w-1$ **down to** 0 **do** ▷ From MSB to LSB of d
3: **if** $d_k = 1$ **then** ▷ k^{th} bit of d
4: $b \leftarrow a \times m \mod n$ ▷ Modular multiplication
5: **else**
6: $b \leftarrow a$
7: **end if**
8: $a \leftarrow b^2 \mod n$ ▷ Modular square
9: **end for**
10: **return** b ▷ $b = m^d \mod n$

Assuming you can input chosen plaintexts, you have access to the output signatures, and you can measure the power consumption during the computations, how would you mount a **power** attack against this implementation?

2 Problem: Fault attacks against DES

Assume you're in charge of attacking a DES implementation. You can send it chosen plaintexts and it will return you the corresponding ciphertexts. You can send the same plaintext several times. Your target is the unknown but constant secret key. During the computation you can inject a fault or not. If you inject a fault, you cannot select when and where. All you know is that your fault causes a single bit flip in the LR state. The bit flip may occur on any of the 64 bits of LR and in any of its 17 successive values (see Figure 1), that is, 1088 possibilities with a uniform probability.

- Propose and explain an attack algorithm to recover the secret key.
- What amount of information can you extract?
- What is the cost of your attack (number of injected faults, number of DES encryption, storage, computations...)?
- Is it practical?
- If yes propose a countermeasure and discuss its efficiency and its drawbacks.

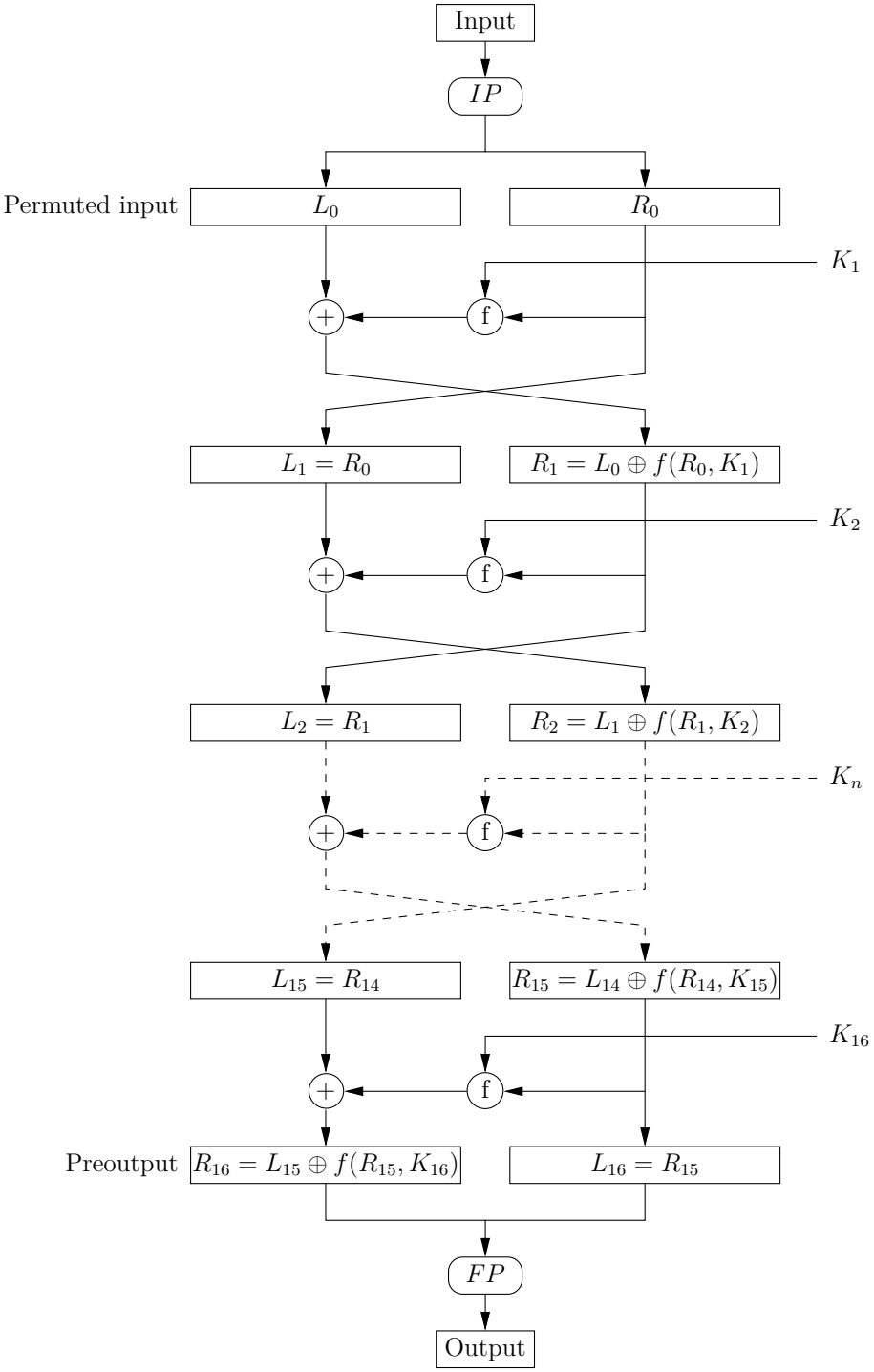

Figure 1: DES encryption