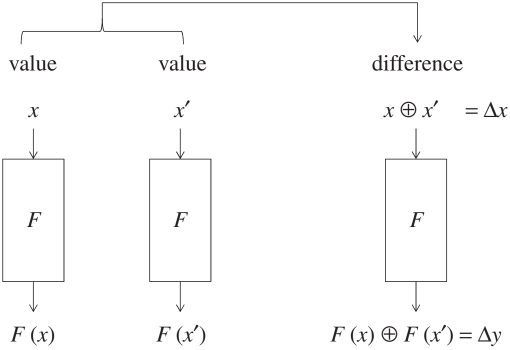
CS 553
CRYPTOGRAPHY

Lecture 8
Automated Differential Cryptanalysis

Instructor
Dr. Dhiman Saha

Image Source: Google

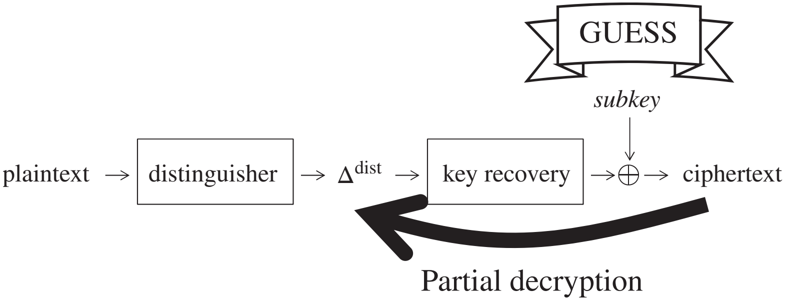
The Notion of Difference



Primary intuition — **Differential Cryptanalysis**

To study the propagation of differences through a cipher focusing on the properties of the Sbox and diffusion layer

Recall — Key Recovery Framework

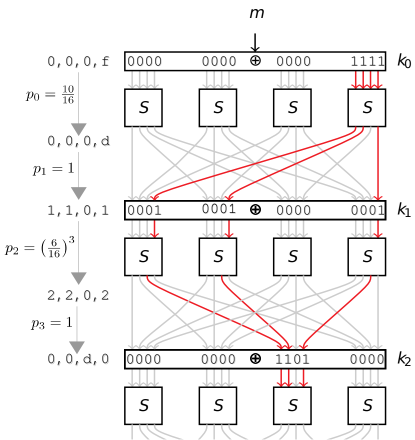


Note

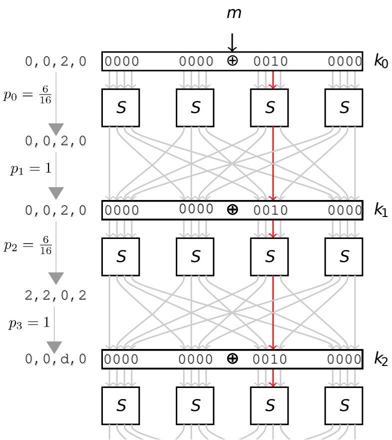
Better distinguisher $\Longrightarrow$ better attack

Image Source: Security of Block Ciphers: From Algorithm Design to Hardware Implementation

Recall: Greedy fails!!! How to search for a good trail?



$$p = \frac{10}{16} \times \left(\frac{6}{16}\right)^3$$



$$p = \left(\frac{6}{16}\right)^2$$

Is There a Way to Automate This in a General Framework?

Computer Aided Cryptanalysis

Introducing Optimization Problem

What is a constrained optimization problem?

Given:

- a set of variables
- an objective function
- a set of constraints
- Find the best solution for the objective function in the set of solutions that satisfy the constraints.

Constraints can be e.g.:

- equations
- inequalities
- linear or non-linear
- restrictions on the type of a variable

Slide inspired from Juila Borghoff's lecture

- It is the study of optimizing (minimizing or maximizing) a **linear** objective function

$$f(x_1, x_2, \cdots, x_n)$$

subject to linear inequalities involving **decision** variables

$$x_i, 1 \leq i \leq n$$

- For many such optimization problems, it is necessary to **restrict** certain decision variables to integer values, i.e. for some values of i, we require $x_i \in \mathbb{Z}$.
- Methods to formulate and solve such programs are called **mixed-integer linear programming (MILP)**.

Let us look at an optimization problem.

```
Minimize
x0 + x1 + x2 + x3 + x4 + x5 + x6 + x7
Subject To
R0: x0 + x1 + x2 + x3 + x4 + x5 + x6 + x7 - 5 d0 >= 0
R1: - x0 + d0 >= 0
R2: - x1 + d0 >= 0
R3: - x2 + d0 >= 0
R4: - x3 + d0 >= 0
R5: - x4 + d0 >= 0
R6: - x5 + d0 >= 0
R7: - x6 + d0 >= 0
R8: - x7 + d0 >= 0
R9: x0 + x1 + x2 + x3 + x4 + x5 + x6 + x7 >= 1
Bounds
Binaries
x0 x1 x2 x3 d0
Generals
x4 x5 x6 x7
End
```

Context of Optimization in Crypto

Crypto problems

- Often described as a set of non-linear Boolean equations
- Algebraic attacks $\implies$ solving non-linear Boolean equations
- Automated solvers often unsuccessful
- Need for new strategies

Optimization

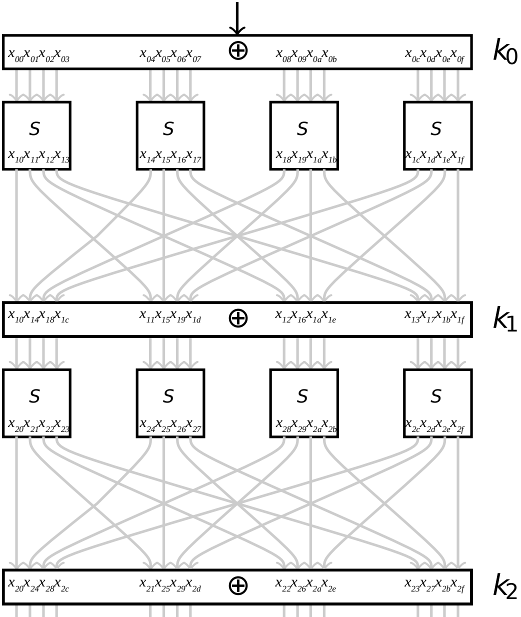
- Well-devolved area
- Many application in operations research
- Algorithms/solver quite evolved
- Many news features available

Slide inspired from Juila Borghoff's lecture

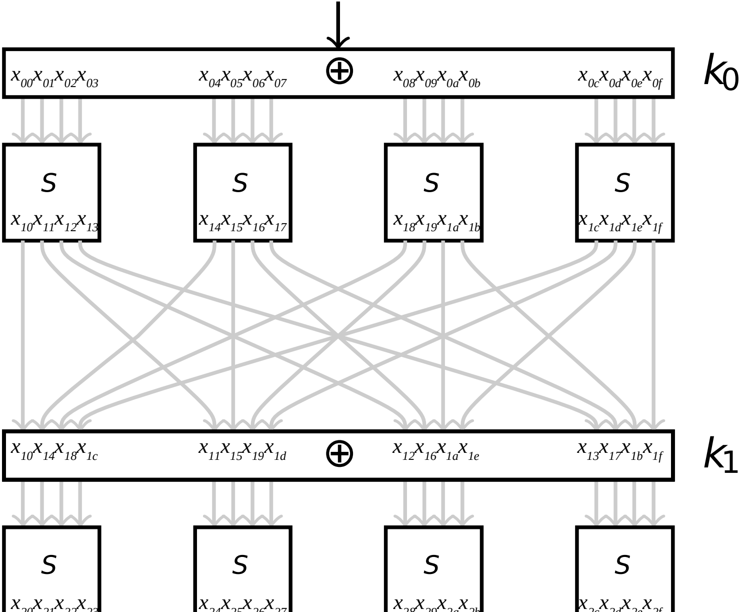
Can we model cryptographic problems as optimization problems?

Modeling Differential Crytanalysis as an Optimization Problem

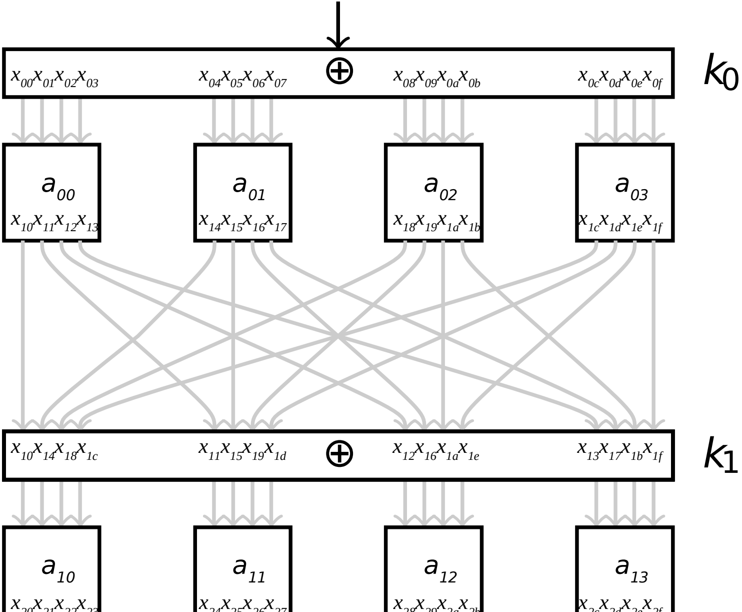
Step-1: $x_{num||bit-pos-hex}$ Bit Variable Naming



Step-1: $x_{num||bit-pos-hex}$ Bit Variable Naming



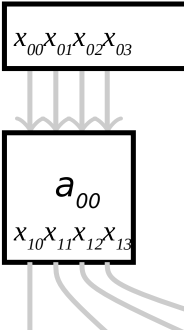
Step-2: $x_{round-num||sbox-pos}$ Sbox Variable Naming



Constraints Describing The Sbox Operation

Firstly, to ensure $a_{ik} = 1$ when any one of x_{ij} in its input is 1.

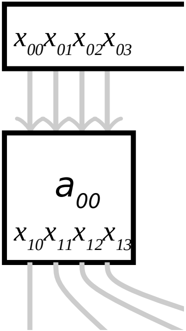
$$x_{00} - a_{00} \leq 0$$
$$x_{01} - a_{00} \leq 0$$
$$x_{02} - a_{00} \leq 0$$
$$x_{03} - a_{00} \leq 0$$



Constraints Describing The Sbox Operation

Secondly, when $a_{ik} = 1$, one of x_{ij} in its input must be 1:

$$x_{00} + x_{01} + x_{02} + x_{03} - a_{00} \geq 0$$



Constraints Describing The Sbox Operation

Thirdly,
input difference must result in output difference and vice versa:

$$4x_{10} + 4x_{11} + 4x_{12} + 4x_{13} - (x_{00} + x_{01} + x_{02} + x_{03}) \geq 0$$
$$4x_{00} + 4x_{01} + 4x_{02} + 4x_{03} - (x_{10} + x_{11} + x_{12} + x_{13}) \geq 0$$