CSL 505

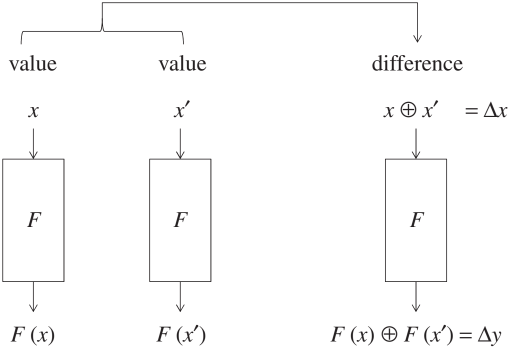
CRYPTOGRAPHY

Lecture 9

Automated Differential Cryptanalysis

Instructor
Dr. Dhiman Saha

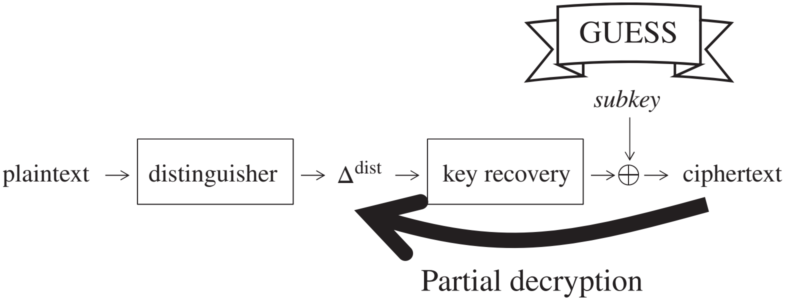
Image Source: Google



Primary intuition — **Differential Cryptanalysis**

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

Key Recovery Framework

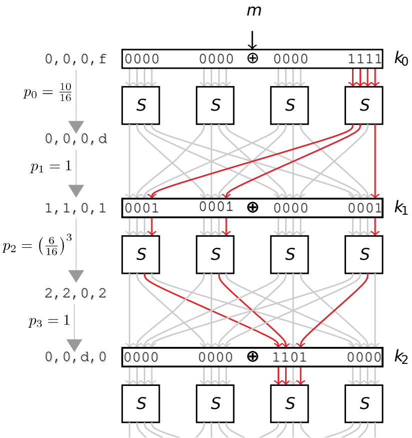


Note

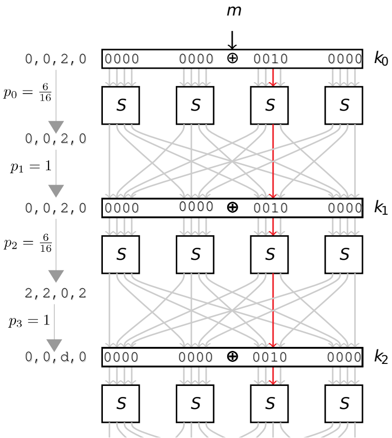
Better distinguisher $\implies$ better attack

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

A simple optimization problem

You become the manager of a small workshop for a day. The workshop produces tables and chairs. Both products consume some amount of wood and labor.

- Total wood available for the day = 100 units
- Total labor available for the day = 80 hours
- Logistics for producing a table
  - Requires 5 units of wood and 2 hours of labor
  - Sells for a profit of ₹400 per table
- Logistics for producing a chair
  - Requires 3 units of wood and 4 hours of labor
  - Sells for a profit of ₹300 per table

Your task is to find out how many tables and chairs to produce so that the profit is maximised

Objective Function, Constraints and Bounds

- The objective function should maximise the profit. Denoting the number of tables produced as x and the number of chairs produced as y, the profit is represented by the following equation

$$400x + 300y$$

- The constraints are on the amount of wood and labor available. The amount of wood used cannot exceed 100 units and the amount of labor used cannot exceed 80 hours.

$$5x + 3y \leq 100$$
$$2x + 4y \leq 80$$

- We also need to ensure that the values of x and y are positive

$$x \geq 0$$
$$y \geq 0$$

```
Maximize
400 x + 300 y
Subject To
R0:  5 x + 3 y <= 100
R1:  2 x + 4 y <= 80
Bounds
0 <= x
0 <= y
Generals
x y
End
```

Commands to run

```
gurobi_cl <filename>.lp
gurobi_cl ResultFile=<output-file>.sol <filename>.lp
```

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 another 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
```

This problem is special and we will revisit it later.

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

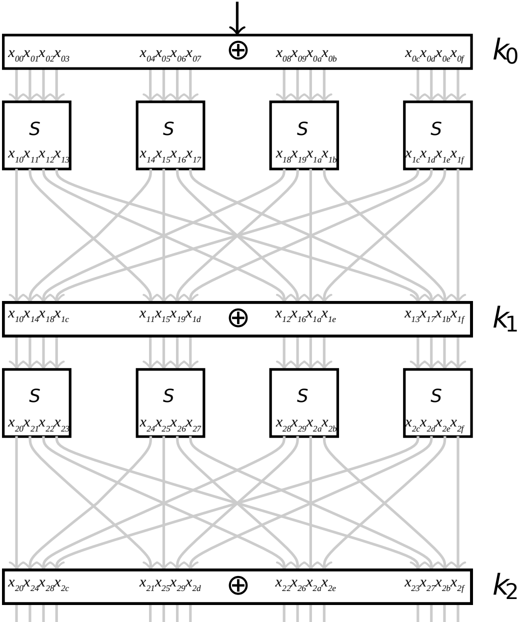
- Well-devolved area
- Many application in operations research
- Algorithms/solver quite evolved
- Many news features available

Slide inspired from Juila Borghoff's lecture

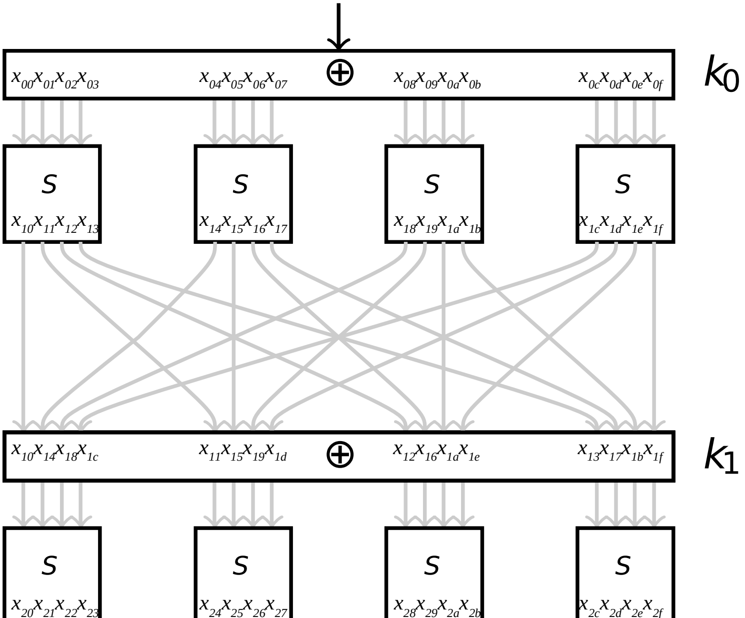
Can we model cryptographic problems as optimization problems?

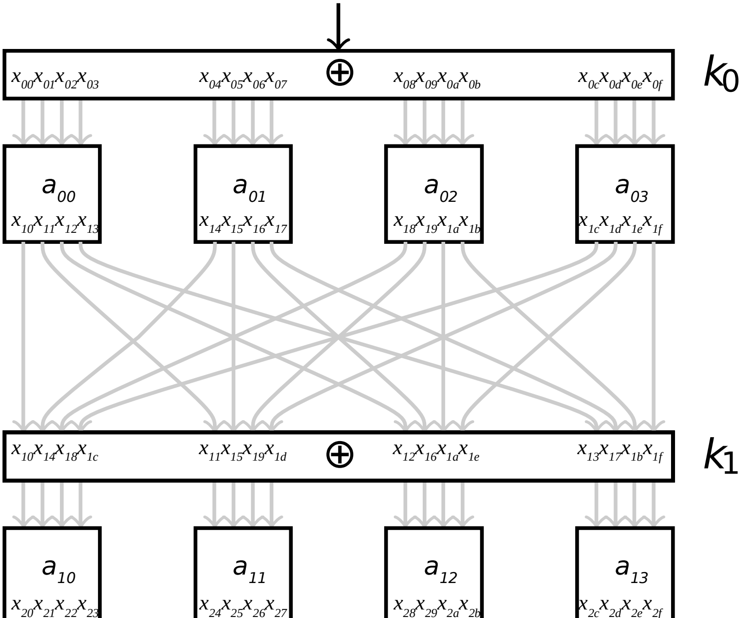
Modeling Differential Crytanalysis as an Optimization Problem

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



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

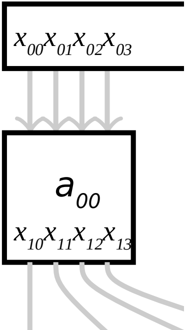



$x_{00}x_{01}x_{02}x_{03}$ $x_{04}x_{05}x_{06}x_{07}$ $\oplus$ $x_{08}x_{09}x_{0a}x_{0b}$ $x_{0c}x_{0d}x_{0e}x_{0f}$
k_0
a_{00} $x_{10}x_{11}x_{12}x_{13}$
a_{01} $x_{14}x_{15}x_{16}x_{17}$
a_{02} $x_{18}x_{19}x_{1a}x_{1b}$
a_{03} $x_{1c}x_{1d}x_{1e}x_{1f}$
$x_{10}x_{14}x_{18}x_{1c}$ $x_{11}x_{15}x_{19}x_{1d}$ $\oplus$ $x_{12}x_{16}x_{1a}x_{1e}$ $x_{13}x_{17}x_{1b}x_{1f}$
k_1
a_{10} $x_{20}x_{21}x_{22}x_{23}$
a_{11} $x_{24}x_{25}x_{26}x_{27}$
a_{12} $x_{28}x_{29}x_{2a}x_{2b}$
a_{13} $x_{2c}x_{2d}x_{2e}x_{2f}$

Constraints Describing The Sbox Operation

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

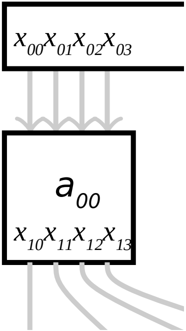
$$x_{00} - a_{00} \leq 0$$
$$x_{01} - a_{00} \leq 0$$
$$x_{02} - a_{00} \leq 0$$
$$x_{03} - a_{00} \leq 0$$



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:

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