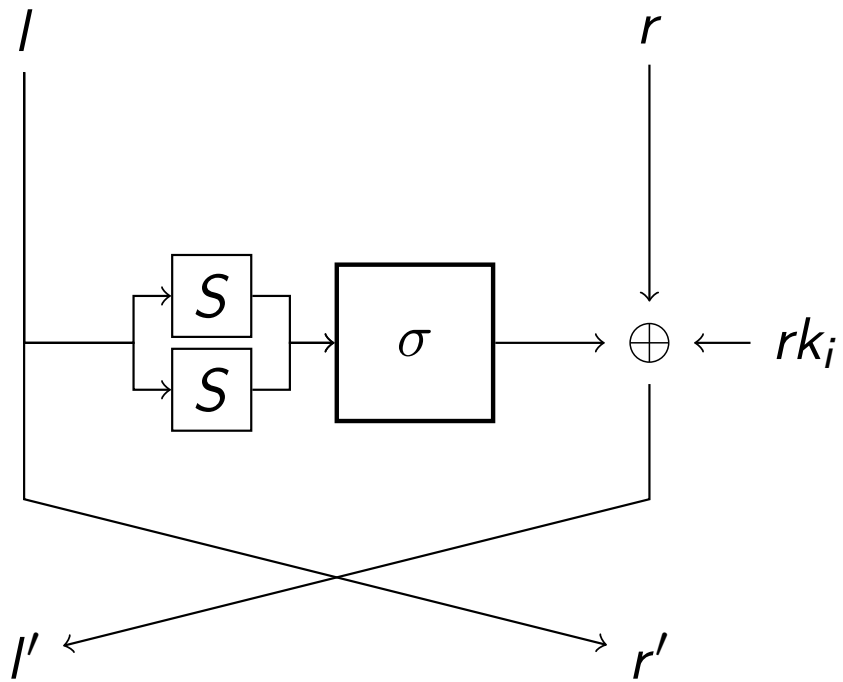
Differential Cryptanalysis

April 3, 2019

Last week's exercise

Solution on whiteboard.

TC05 (mini)



With Sbox:

$$S = (\texttt{E}, \texttt{B}, \texttt{4}, \texttt{6}, \texttt{A}, \texttt{B}, \texttt{7}, \texttt{0}, \texttt{3}, \texttt{8}, \texttt{F}, \texttt{C}, \texttt{5}, \texttt{9}, \texttt{1}, \texttt{2})$$

and bit permutation:

$$\sigma = \left(\begin{array}{cccc|cccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 6 & 0 & 1 & 7 & 2 & 4 & 5 & 3 \end{array}\right)$$

Difference propagation

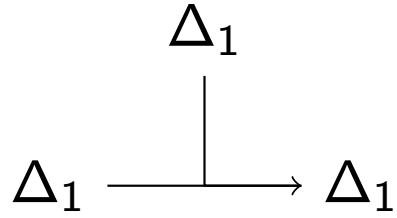


Figure: Branching

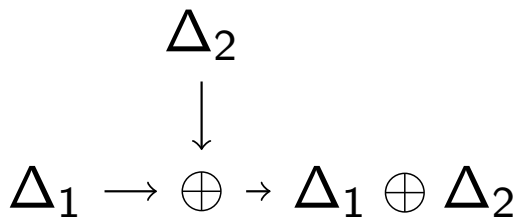


Figure: Xor

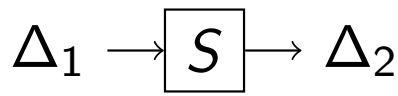


Figure: S-box

Differences

Take the TC05 sbox:

$$S = (\texttt{E}, \texttt{B}, \texttt{4}, \texttt{6}, \texttt{A}, \texttt{B}, \texttt{7}, \texttt{0}, \texttt{3}, \texttt{8}, \texttt{F}, \texttt{C}, \texttt{5}, \texttt{9}, \texttt{1}, \texttt{2})$$

Questions

- What is $S(1)$, $S(2)$ and $S(3)$?
- Is $S(1 \oplus 2) = S(1) \oplus S(2)$?
- If you get an input difference of `0x4`, which output difference is most likely?
- Would you use this sbox as a building block for a cipher?

Difference Distribution Table

- A useful tool for differential cryptanalysis is the DDT
- Computing the DDT for small permutations is trivial
- Naive Time complexity is 2^{3n}, for a n bit permutation
- You can implement it yourself or use Sage (`https://www.sagemath.org/`).

Algorithm 1 Compute_$DDT(sbox)$

Let DDT be a $n \times n$ array
for $\Delta_{in} \in \mathbb{F}_2^n$ **do**
  for $p \in \mathbb{F}_2^n$ **do**
    $\Delta_{out} = \sigma(p) \oplus \sigma(p \oplus \Delta_{in})$
    $DDT[\Delta_{in}][\Delta_{out}]$
  end for
end for
return DDT

TC05 sbox DDT

	0	1	2	3	4	5	6	7	8	9	*A*	*B*	*C*	*D*	*E*	*F*
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	2	4	0	2	0	4	0	0	0	2	2	0	0	0
2	0	0	0	0	4	0	0	0	0	0	2	2	2	6	0	0
3	0	0	0	0	0	0	0	4	4	0	4	0	0	0	0	4
4	0	2	0	2	2	0	6	0	0	0	0	0	0	0	4	0
5	0	4	0	2	2	0	0	0	0	0	2	0	0	6	0	0
6	0	0	2	0	0	2	0	0	0	2	4	4	0	0	2	0
7	0	2	0	0	0	0	2	0	0	6	0	0	4	0	2	0
8	0	0	2	2	2	0	2	0	0	0	2	2	0	2	0	2
9	0	2	0	2	0	2	2	0	6	2	0	0	0	0	0	0
A	0	2	2	0	0	0	0	4	0	2	0	2	0	0	2	2
B	0	0	2	0	2	4	0	0	2	0	0	0	4	0	2	0
C	0	0	2	0	2	4	0	0	2	2	0	2	2	0	0	0
D	0	0	2	0	0	0	2	4	0	0	0	2	0	0	4	2
E	0	4	0	0	2	2	0	0	2	2	0	0	0	0	0	4
F	0	0	2	4	0	0	2	0	0	0	2	0	2	2	0	2

TC05 sbox DDT

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	2	4	0	2	0	4	0	0	0	2	2	0	0	0
2	0	0	0	0	4	0	0	0	0	0	2	2	2	6	0	0
3	0	0	0	0	0	0	0	4	4	0	4	0	0	0	0	4
4	0	2	0	2	2	0	6	0	0	0	0	0	0	0	4	0
5	0	4	0	2	2	0	0	0	0	0	2	0	0	6	0	0
6	0	0	2	0	0	2	0	0	0	2	4	4	0	0	2	0
7	0	2	0	0	0	0	2	0	0	6	0	0	4	0	2	0
8	0	0	2	2	2	0	2	0	0	0	2	2	0	2	0	2
9	0	2	0	2	0	2	2	0	6	2	0	0	0	0	0	0
A	0	2	2	0	0	0	0	4	0	2	0	2	0	0	2	2
B	0	0	2	0	2	4	0	0	2	0	0	0	4	0	2	0
C	0	0	2	0	2	4	0	0	2	2	0	2	2	0	0	0
D	0	0	2	0	0	0	2	4	0	0	0	2	0	0	4	2
E	0	4	0	0	2	2	0	0	2	2	0	0	0	0	0	4
F	0	0	2	4	0	0	2	0	0	0	2	0	2	2	0	2

DDT

Questions

- What is the differential uniformity of TC05's sbox?
- If we have an input difference of 7, what is the probability that we see an output difference of 4?
- What does each column and row sum up to? and why?

Differential cryptanalysis

- We try to find an input difference $\Delta_{\mathtt{in}}$ such that the probability of getting a certain $\Delta_{\mathtt{out}}$ after r rounds is 'high'
- Now we have an r round distinguisher, that can distinguish the cipher from ia random permutation.
- Plan: get pairs with certain $\Delta_{\mathtt{in}}$, decrypt i rounds to reach the r-th round and if we see the $\Delta_{\mathtt{out}}$: Profit!
- Few technicalities, but we will deal with that.

Implementing Differential attacks

- Two parts: Key recovery and Distinguisher
- Distinguisher only needs to be found: by hand/computer.
- Key recovery needs a bit more work.

Key recovery (naive)

- Prepending rounds
  - Given an r round distinguisher from $\Delta_{\mathtt{in}} \rightarrow \Delta_{\mathtt{out}}$ we can attack $r + i$ rounds and a set of plaintext ciphertext pairs P by partially encrypting i rounds.
    - Guess partial key k s.t. we can decrypt i rounds.
    - Initialize a counter for k to 0
    - Construct the set $P' = (E_i(p, k), c)$ for $(p, c) \in P$.
    - Now for every pair $(p_1, c_1), (p_2, c_2) \in P'$
    - If $p_1 \oplus p_2 = \Delta_{\mathtt{in}} \wedge c_1 \oplus c_2 = \Delta_{\mathtt{out}}$ increase the counter for key k by one.
  - Pick the key k with the highest counter.
- Appending rounds analogous to prepending. (Often works better than prepending)

Questions

- In what attacker model lies Differential Cryptanalysis?
- Given a cipher with blocksize n what is the minimum probability of a characterstic that we can use.
- What is the most important component that guards against differential cryptanalysis?
- Why is appending rounds for key recovery easier than prepending rounds?
- What are the drawbacks of Differential Cryptanalysis w.r.t. the other attacks we discussed in this course?

Next week

- Do exercise for this week.
- Read the (extra) material on the website.
- Catch up on previous exercises.
- If you have questions, I'll be in the office from Sunday, otherwise email me.