TAGADA:
Tool for Automatic Generation of Abstraction-based Differential Attacks

Ana Margarita Rodríguez Cordero

Université de Lorraine, CNRS, Inria, LORIA, Nancy, France
Université Clermont Auvergne, LIMOS

April 14, 2022

Based on work with:
Luc Libralesso – François Delobel – Pascal Lafourcade – Christine Solnon
Symmetric Cryptography

An example of a symmetric key system is shown in the diagram: a secret key is used for both encryption and decryption. The process involves:

1. **Encryption**: The original document is transformed into ciphertext using the secret key. For example:
   - Original document
   - Secret key
   - Encrypted document

2. **Decryption**: The ciphertext is transformed back into the original document using the same secret key. For example:
   - Ciphertext
   - Secret key
   - Decrypted document

This diagram illustrates the symmetric nature of the key used in both encryption and decryption processes.
Symmetric Cryptography

- Stream ciphers
- Block ciphers
Symmetric Cryptography

- Block ciphers
Definition

Given a key $K \in \mathbb{F}_2^m$, a message $M \in \mathbb{F}_2^N$, a \textit{block cipher} of block size $n$ is an invertible function $E_K$ that encrypts the message $M$ in blocks of size $n$. 

Block cipher: $E_K$

\[ \text{Plaintext} \rightarrow E_K \rightarrow \text{Ciphertext} \]

Key

Block cipher: $E_K^{-1}$

\[ \text{Ciphertext} \rightarrow E_K^{-1} \rightarrow \text{Plaintext} \]
SPN and Feistel cipher

SP Network

Feistel Structure

\[
\begin{array}{cccccccc}
\times & 0x0 & 0x1 & 0x2 & 0x3 & 0x4 & 0x5 & 0x6 & 0x7 \\
S(x) & 0x5 & 0x3 & 0x4 & 0x6 & 0x2 & 0x7 & 0x0 & 0x1
\end{array}
\]
Cryptanalysis

- Can we distinguish the cipher from a random permutation?
- Is the ciphertext giving us any information?
- Is there any weakness in the design?
Cryptanalysis

- Can we distinguish the cipher from a random permutation?
- Is the ciphertext giving us any information?
- Is there any weakness in the design?

Attack techniques:
- ciphertext-only
- known plaintext
- chosen plaintext
Cryptanalysis

- Can we distinguish the cipher from a random permutation?
- Is the ciphertext giving us any information?
- Is there any weakness in the design?

Attack techniques:
- ciphertext-only
- known plaintext
- chosen plaintext

Type of attacks:
- Differential attack
- Boomerang attack
- Linear attack
- Square attack
- ...
Cryptanalysis

- Can we distinguish the cipher from a random permutation?
- Is the ciphertext giving us any information?
- Is there any weakness in the design?

Attack techniques:
- ciphertext-only
- known plaintext
- chosen plaintext

Type of attacks:
- Differential attack
- Boomerang attack
- Linear attack
- Square attack
- ...
Differential Attacks

\[ x_0 \oplus x'_0 = \Delta_0 = \Delta \]

\[ x_1 \oplus x'_1 = \Delta_1 \]

\[ \ldots \oplus \ldots = \Delta_r = \nabla \]

- \( \Delta \) - input difference
- \( \nabla \) - output difference
- \( \nabla = E_K(X) \oplus E_K(\Delta \oplus X) \), for \( X \in \mathbb{F}_2^n \)
- Is \( P(\Delta \rightarrow \nabla) \) high?
Differential Attacks

\[ x_0 \oplus x'_0 = \Delta_0 = \Delta \]

\[ x_1 \oplus x'_1 = \Delta_1 \]

\[ \cdots \oplus \cdots \oplus \cdots \]

\[ x_r \oplus x'_r = \Delta_r = \nabla \]

- $\Delta$ - input difference
- $\nabla$ - output difference
- $\nabla = E_K(X) \oplus E_K(\Delta \oplus X)$, for $X \in \mathbb{F}_2^n$
- Is $P(\Delta \rightarrow \nabla)$ high?

Related-key differential attack: Differentials are also introduced in the key.

$\nabla = E_K(X) \oplus E_{\Delta K \oplus K}(\Delta \oplus X)$
Differential Attack

- Step 1: *Abstraction*:
  - Truncated differential patterns.
  - Number of S-boxes minimized.
Differential Attack

- **Step 1: Abstraction:**
  - Truncated differential patterns.
  - Number of S-boxes minimized.
- **Step 2: Enumeration:**
  - Find nonabstracted differential characteristics: Distinguishers.
Step 1

Find minimum number of active S-boxes

Find all difference patterns minimizing the active number of S-boxes

Ana Margarita Rodríguez Cordero
Step 1

- Find minimum number of active S-boxes
Step 1

- Find minimum number of active S-boxes
- Find all difference patterns minimizing the active number of S-boxes

\( \Delta_i \)
\[
\begin{array}{cccccc}
\Delta_i & 7 & 0 & 0 & 2 & 0 & 3 \\
S_1 & 5 & 0 & 0 & 4 & 0 & 6 \\
S_2 & 0 & 0 & 0 & 5 & 6 & 0 \\
S_3 & 4 & 0 & 0 & 0 & 0 & 0 \\
S_4 & 0 & 1 & 0 & 1 & 0 & 1 \\
S_5 & 1 & 0 & 1 & 1 & 0 & 0 \\
S_6 & 0 & 1 & 1 & 1 & 1 & 0 \\
\end{array}
\]

\( T \Delta_i \)
\[
\begin{array}{cccccc}
T \Delta_i & 1 & 0 & 0 & 1 & 0 & 1 \\
S_1 & 1 & 0 & 0 & 1 & 0 & 1 \\
S_2 & 0 & 0 & 0 & 0 & 1 & 0 \\
S_3 & 0 & 1 & 1 & 1 & 0 & 0 \\
S_4 & 1 & 0 & 0 & 1 & 0 & 0 \\
S_5 & 0 & 1 & 1 & 1 & 0 & 0 \\
S_6 & 1 & 0 & 0 & 1 & 0 & 0 \\
\end{array}
\]
Step 2

- Find a differential characteristic that fits the truncated pattern.
- Modelling the S-box Difference Distribution Table through constraint programming:
  - MILP modelling
  - SAT modelling
Step 2

- Find a differential characteristic that fits the truncated pattern.
- Modelling the S-box Difference Distribution Table through constraint programming:
  - MILP modelling
  - SAT modelling

**Difference Distribution Table:**

\[ DDT(\Delta_i, \nabla_o) = \# \{ x \in \mathbb{F}_2^n : S(x) \oplus S(x \oplus \Delta_i) = \nabla_o \} \]

<table>
<thead>
<tr>
<th>(\Delta_i)</th>
<th>0x0</th>
<th>0x1</th>
<th>0x2</th>
<th>0x3</th>
<th>0x4</th>
<th>0x5</th>
<th>0x6</th>
<th>0x7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x0</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0x1</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>0x2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>0x3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>0x4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>0x5</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>0x6</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>0x7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Objective:

Obtain a good differential characteristic for any cipher given in input.
Objective:
Obtain a good differential characteristic for any cipher given in input

- Give a cipher description
Objective:

Obtain a good differential characteristic for any cipher given in input

- Give a cipher description
- Run TAGADA
Objective:

Obtain a good differential characteristic for any cipher given in input

- Give a cipher description
- Run TAGADA
- Obtain an attack
TAGADA’s idea

cipher $\rightarrow$ DAG

cipher $\rightarrow$ DAG $\rightarrow$ Abstraction $\rightarrow$ Step 1 $\rightarrow$ Active S-boxes

Step 2 $\leftrightarrow$ Truncated characteristics

cipher $\rightarrow$ DAG $\rightarrow$ Abstraction $\rightarrow$ Step 1 $\rightarrow$ Active S-boxes

Step 2 $\leftrightarrow$ Differential characteristics

test vectors $\rightarrow$ Differential characteristics
**Cipher oriented language**

- **State**: Internal state of the cipher at a given time (integer variable).
- **Operator**: Block used for changing from one state to another.
```python
def create_cherry_dag(nb_rounds):
    # define dag and inputs
    dag = Dag.new([], [], [], [])
    x, _, _ = dag.register_block(*input_block("X", NIBBLE_RANGE, [1,4]))
    dag.set_plaintexts(x.flatten)
    k, _, _ = dag.register_block(*input_block("K", NIBBLE_RANGE, [1,4]))
    dag.set_inputs(x.flatten+k.flatten)
    dag.set_keys(k.flatten)
    a = x
    b = k
    nb_rounds.times do |round_number|
        # ARK
        a, _, _ = dag.register_block(*xor_block("ARK_#{round_number}", [a,b]))
        # pLayer
        a, _, _ = dag.register_block(*permutation_block("P_#{round_number}", a,pLayer))
        # ShiftRows
        a, _, _ = dag.register_block(*shiftrows_block("SR_#{round_number}", a,false))
        # S-box
        a, _, _ = dag.register_block(*subcell_block("S_#{round_number}", a, sbox))
        ############### key update
        b, _, _ = dag.register_block(*permutation_block("KU_#{round_number}", b, pKey))
    end
    dag.set_outputs(a.flatten)
    return dag
end
```

Figure: Toy cipher
Shaving

Figure: AES Directed Acyclic Graph
Figure: AES shaved Directed Acyclic Graph
## Operators

Operator description $\rightarrow$ Table of constraints

<table>
<thead>
<tr>
<th>$a$</th>
<th>$b$</th>
<th>$a \oplus b$</th>
<th>abstraction($a,b,a \oplus b$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(0,0,0)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>(0,1,1)</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>255</td>
<td>254</td>
<td>1</td>
<td>(1,1,1)</td>
</tr>
<tr>
<td>255</td>
<td>255</td>
<td>0</td>
<td>(1,1,0)</td>
</tr>
</tbody>
</table>

$(0,0,0), (0,1,1), (1,0,1), (1,1,0), (1,1,1) \Rightarrow a + b + XOR(a, b) \neq 1$
Results in AES

Round function $f$

$C \leftarrow M \times C$

$w_i \leftarrow M \times C$

$z_i \leftarrow S \times R$

$y_i \leftarrow S \times R$

$x_i \leftarrow A \times K$

$w_{i-1}$

$\mathbf{S}$
Results in SKINNY

**Figure:** SKINNY round function

![SKINNY round function diagram](image_url)
## Results in SKINNY

<table>
<thead>
<tr>
<th>ROUNDS</th>
<th>State-of-the-art</th>
<th>TAGADA</th>
<th>State-of-the-art</th>
<th>TAGADA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Sb</td>
<td>#Sol</td>
<td>Time</td>
<td>#Sb</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>4</td>
<td>1s</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>3</td>
<td>1s</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
<td>2</td>
<td>1s</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>1</td>
<td>1s</td>
<td>16</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>4</td>
<td>1s</td>
<td>26</td>
</tr>
<tr>
<td>8</td>
<td>36</td>
<td>17</td>
<td>1s</td>
<td>36</td>
</tr>
<tr>
<td>9</td>
<td>41</td>
<td>2</td>
<td>1s</td>
<td>41</td>
</tr>
<tr>
<td>10</td>
<td>46</td>
<td>2</td>
<td>1s</td>
<td>49</td>
</tr>
<tr>
<td>11</td>
<td>51</td>
<td>2</td>
<td>1s</td>
<td>51</td>
</tr>
<tr>
<td>12</td>
<td>55</td>
<td>2</td>
<td>1s</td>
<td>55</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>6</td>
<td>1s</td>
<td>58</td>
</tr>
<tr>
<td>14</td>
<td>61</td>
<td>2</td>
<td>1s</td>
<td>61</td>
</tr>
<tr>
<td>15</td>
<td>66</td>
<td>2</td>
<td>1s</td>
<td>66</td>
</tr>
<tr>
<td>16</td>
<td>75</td>
<td>8</td>
<td>1s</td>
<td>75</td>
</tr>
<tr>
<td>17</td>
<td>82</td>
<td>4</td>
<td>1s</td>
<td>82</td>
</tr>
<tr>
<td>18</td>
<td>88</td>
<td>4</td>
<td>1s</td>
<td>-</td>
</tr>
<tr>
<td>19</td>
<td>92</td>
<td>4</td>
<td>1s</td>
<td>-</td>
</tr>
<tr>
<td>20</td>
<td>96</td>
<td>2</td>
<td>1s</td>
<td>-</td>
</tr>
<tr>
<td>21</td>
<td>105</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>109</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>113</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Results in WARP

### Related key results for WARP

<table>
<thead>
<tr>
<th>Round</th>
<th>S-boxes</th>
<th>N-sol</th>
<th>Time in Gurobi (s)</th>
<th>Round</th>
<th>S-boxes</th>
<th>N-sol</th>
<th>Time in Gurobi (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>16</td>
<td>0.0927</td>
<td>23</td>
<td>11</td>
<td>16</td>
<td>30.7</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>32</td>
<td>0.0901</td>
<td>24</td>
<td>12</td>
<td>16</td>
<td>29.7</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>48</td>
<td>0.0997</td>
<td>25</td>
<td>12</td>
<td>16</td>
<td>35.0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>64</td>
<td>0.103</td>
<td>26</td>
<td>13</td>
<td>16</td>
<td>37.7</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>96</td>
<td>0.122</td>
<td>27</td>
<td>13</td>
<td>16</td>
<td>49.9</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>72</td>
<td>0.618</td>
<td>28</td>
<td>14</td>
<td>36</td>
<td>116</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>16</td>
<td>1.19</td>
<td>29</td>
<td>15</td>
<td>36</td>
<td>42.3</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>36</td>
<td>1.26</td>
<td>30</td>
<td>16</td>
<td>36</td>
<td>71.3</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>16</td>
<td>2.02</td>
<td>31</td>
<td>15</td>
<td>16</td>
<td>64.4</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>36</td>
<td>2.69</td>
<td>32</td>
<td>16</td>
<td>36</td>
<td>77.0</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>16</td>
<td>3.87</td>
<td>33</td>
<td>16</td>
<td>16</td>
<td>79.8</td>
</tr>
<tr>
<td>12</td>
<td>6</td>
<td>36</td>
<td>5.31</td>
<td>34</td>
<td>17</td>
<td>36</td>
<td>76.2</td>
</tr>
<tr>
<td>13</td>
<td>6</td>
<td>16</td>
<td>6.27</td>
<td>35</td>
<td>17</td>
<td>16</td>
<td>113</td>
</tr>
<tr>
<td>14</td>
<td>7</td>
<td>36</td>
<td>10.5</td>
<td>36</td>
<td>18</td>
<td>36</td>
<td>111</td>
</tr>
<tr>
<td>15</td>
<td>7</td>
<td>16</td>
<td>8.12</td>
<td>37</td>
<td>18</td>
<td>16</td>
<td>241</td>
</tr>
<tr>
<td>16</td>
<td>8</td>
<td>36</td>
<td>11.2</td>
<td>38</td>
<td>19</td>
<td>36</td>
<td>360</td>
</tr>
<tr>
<td>17</td>
<td>8</td>
<td>16</td>
<td>15.9</td>
<td>39</td>
<td>19</td>
<td>16</td>
<td>81.7</td>
</tr>
<tr>
<td>18</td>
<td>9</td>
<td>36</td>
<td>15.9</td>
<td>40</td>
<td>20</td>
<td>36</td>
<td>148</td>
</tr>
<tr>
<td>19</td>
<td>9</td>
<td>16</td>
<td>14.5</td>
<td>41</td>
<td>20</td>
<td>16</td>
<td>218e</td>
</tr>
<tr>
<td>20</td>
<td>10</td>
<td>36</td>
<td>28.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>10</td>
<td>16</td>
<td>30.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>11</td>
<td>36</td>
<td>24.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Conclusion

- Tagada works with word-based ciphers.
Conclusion

- **Tagada** works with word-based ciphers.
- Ciphers can be described as graphs with **Tagada**.
**Conclusion**

- **TAGADA** works with word-based ciphers.
- Ciphers can be described as graphs with **TAGADA**.
- We obtain good resolution times compare with the state-of-the-art attacks.
**Conclusion**

- **Tagada** works with word-based ciphers.
- Ciphers can be described as graphs with **Tagada**.
- We obtain good resolution times compared with the state-of-the-art attacks.

<table>
<thead>
<tr>
<th>GIFT</th>
<th>PRESENT</th>
<th>HIGHT</th>
<th>GIFT-COFB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deoxys</td>
<td>SKINNY</td>
<td>ISAP</td>
<td>Sparkle</td>
</tr>
<tr>
<td>WARP</td>
<td>CRAFT</td>
<td>Midori</td>
<td>Photon</td>
</tr>
<tr>
<td>Rijndael</td>
<td>Simeck</td>
<td>Mysterion</td>
<td>Elephant</td>
</tr>
</tbody>
</table>
References

Source code:
https://gitlab.limos.fr/iia_lulibral/tagada/

Luc Libralesso and François Delobel and Pascal Lafourcade and Christine Solnon
“Automatic Generation of Declarative Models For Differential Cryptanalysis”.
Thanks for your attention!