Algebraic Immunity of S-boxes and Augmented Functions

Simon Fischer and Willi Meier

University of Applied Sciences
Northwestern Switzerland

Outline

Part 1
Algebraic Properties of S-boxes

The S-box

Notation: F denotes GF(2), and S is the S-box

$$S : F^n \to F^m$$

Input $x = (x_1, \ldots, x_n)$, output $y = (y_1, \ldots, y_m)$, and $S(x) = y$.

Scenario: y is known, recover x with algebraic equations.

Use equations conditioned by some fixed y: **conditional equations** (CE).
These are equations in x, which holds for all preimages of some y.
Can find optimal equation (minimum degree) for each y (Armknecht).

How to Find Conditional Equations

Use matrix approach to find CE's (Courtois).

Example: S-box with $n = 3$, assume some output y with preimages $x = 100, 110, 011, 001$. Find linear CE.

$$
\begin{array}{cccccc|c}
 & 1 & x_1 & x_2 & x_3 & & \text{preimages} \\
\hline
 & 1 & 1 & 0 & 0 & & x = 100 \\
M = & 1 & 1 & 1 & 0 & & x = 110 \\
 & 1 & 0 & 1 & 1 & & x = 011 \\
 & 1 & 0 & 0 & 1 & & x = 001
\end{array}
$$

$$M = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Solution: $0 = 1 + x_1 + x_3$ holds for each preimage.

How to Find Conditional Equations

Use matrix approach to find CE's (Courtois).

Example: S-box with $n = 3$, assume some output y with preimages $x = 100, 110, 011, 001$. Find linear CE.

$$\begin{array}{c|c} \begin{array}{cccc} 1 & x_1 & x_2 & x_3 \end{array} & \text{preimages} \\ \hline M = \begin{pmatrix} \boxed{1} & \boxed{1} & 0 & \boxed{0} \\ \boxed{1} & \boxed{1} & 1 & \boxed{0} \\ \boxed{1} & \boxed{0} & 1 & \boxed{1} \\ \boxed{1} & \boxed{0} & 0 & \boxed{1} \end{pmatrix} & \begin{array}{c} x = 100 \\ x = 110 \\ x = 011 \\ x = 001 \end{array} \end{array}$$

Solution: $0 = 1 + x_1 + x_3$ holds for each preimage.

Theoretical Background

Number of preimages: 2^{n-m} for balanced S-box.

Number of monomials: $D = \sum_{i=0}^{d} \binom{n}{i}$ for degree d.

Matrix M has 2^{n-m} rows, and D columns.

Number of CE's corresponds to the dimension of solution space of M.

Sufficient condition for existence of CE: $2^{n-m} < D$.
If m is parameter: $m > m_0$ with $m_0 := n - \log_2 D$.

Weak output: CE exists though $m \ll m_0$.

Algorithmic Methods

Can find CE's by setting up and solving M.
Bottleneck: finding all preimages takes 2^n steps.

Probabilistic algorithm:

- A random preimage can be found in 2^m.
- Solve smaller matrix M with a few **random preimages**.
- If CE exists, it holds only for fraction p of all 2^{n-m} preimages.
- With about D random preimages, p will be very large.

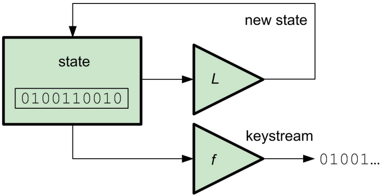
Complexity is $2^m D + D^3$.

Probabilistic algorithm is efficient for weak outputs.

Part 2
Augmented Functions

Situation

Stream cipher with update function L, output function f.
Update L is linear (e.g. in LFSR) or nonlinear (e.g. in Trivium).



S-box in context of stream cipher: **augmented function** (AF).

$$\begin{aligned} S_m : F^n &\to F^m \\ x &\mapsto (f(x), f(L(x)), \dots, f(L^{m-1}(x))) \end{aligned}$$

New Scenarios of Algebraic Attacks

Use probabilistic algorithm to find CE's for AF, recover x.

Block size:
m is a natural parameter for augmented function S_m.

Finding preimages:
In 2^m for random S-box. AF can have simple structure.
Sampling methods in TMTO attacks (Biryukov-Shamir).

New algebraic attacks on AF, if:

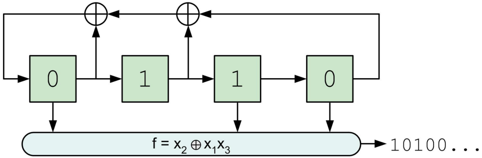
1. AF has many weak outputs (low-degree CE's for $m \ll m_0$).
2. Finding preimages is feasible (for output size m).

Part 3

Application: Filter Generators

Situation

LFSR of n bits, and Boolean function f.



Algebraic Attacks:

- f has algebraic immunity e, linearisation requires $\binom{n}{e}$ data.
- Gröbner bases need only about n bit data in few cases (experimental results by Faugère-Ars).

Understand such behavior with augmented function.

Existence of Equations

Experiments:
Consider CanFil family (as in Faugère-Ars) and Majority function.
State of size $n = 20$, find linear equations where $m_0 = 16$.

Step 1: Existence of exact equations (by computing all preimages)

Example

$n = 20$, fixed setup, CanFil5 $= x_1 + x_2 x_3 + x_2 x_3 x_4 x_5$.
Output $y = 000000$ of $m = 6$ bits.
There are 2^{14} preimages, and $D = 21$ monomials in matrix M.
M has rank 20, one linear equation exists.

The output $y = 000000$ seems very weak. What about other outputs?
What about other setups and functions?

Exact Equations

For $n = 20$, record overall number of equations (for all y):

Filter	m	Different setups				
CanFil1	14	0	0	0	0	0
	15	3139	4211	3071	4601	3844
CanFil2	14	0	0	0	0	0
	15	2136	2901	2717	2702	2456
CanFil5	6	0	0	0	2	0
	7	0	0	0	8	0
	8	0	0	0	24	0
	9	0	0	0	64	0
	10	6	0	0	163	0
	11	113	0	2	476	0
	12	960	16	215	1678	29
Majority5	9	0	0	0	2	0
	10	1	10	1	18	1
	11	22	437	40	148	56

Linear equations exist only for m about m_0.

Linear equations exist already for m about $n/2$.

Observation 1: Number of equations mainly depends on filter function.
Observation 2: Experimental results are scalable with n.

Probabilistic Equations

Try to find equations with the probabilistic algorithm.

Step 2: Probabilistic equations (by computing a few random preimages)

Example

$n = 20$, fixed setup, CanFil5, $y = 000000$ of $m = 6$ bits.
Pick instead of all 2^{14} preimages only $N = 80$ random preimages, $D = 21$.
Determine all solutions for much smaller matrix M.
Obtained always 2 to 4 solutions, with probability $p = 0.98, \ldots, 1$.

Probability impressively large $\rightarrow$ probabilistic equations useful in attacks.

Sampling

Step 3: Sampling (efficient computation of random preimages)

Filter inversion:
Fix k inputs of filter which give correct observed output bit.
Repeat for about n/k output bits, until state is unique.
Complexity $2^{m-n/k}$ to find one preimage, efficient if k is small.

Linear sampling:
Impose linear conditions on input variables, so that f becomes linear.
Solve linear system to find one preimage.

With sampling, can find equations for quite large n.
Example with CanFil5, $n = 80$, $m = 40$. Linear equation in 2^{32} for some y.

Algebraic Attacks

Each new low degree equation (found by investigating AF) can serve to reduce data complexity of algebraic attacks.

Have identified functions f which show **resistance** to this approach:
Equations exist only for large m, effort of finding preimages is too large.

Several other functions f shown to be **weak**:
Many low degree equations can be determined efficiently.

In some cases, data complexity can be of order n:
Observe n weak outputs and set up n linear equations.

Part 4
Application: Trivium

Sampling

State of $n = 288$ bits, nonlinear update, linear output of one bit.

Consider AF with n input bits and m consecutive output bits.
Use our framework, but how to find preimages for such a large state?

Sampling:
In first 66 clocks, each keystream bit is linear in initial state bits.
Finding preimages for $m = 66$ obvious.

For larger m, use linear sampling:
Fix even bits of state, get linear relations in remaining variables.
Can find preimages efficiently for $m = n/2 = 144$ or larger.

Experimental Results

Are there additional linear equations beyond the 66 known ones?

Example

Consider AF of Trivium with $m = 144$.
Choose random output y and find $N = 400$ preimages.
Set up and solve matrix M with N preimages and $D = 289$ monomials.

Result: For different y, get always 66 linear equations.

Can go further: Determine preimages for $m = 150$ with partial search.
Still find 66 linear equations for a 150 bit output of consecutive 0's.

Trivium seems resistant against additional linear equations in AF.

Conclusions

Conclusions

1. The augmented function of a stream cipher should be checked for conditional equations of low degree.
2. This requires computation of preimages, can be efficient in some cases.
3. Checking successful for a class of filter generators and for Trivium.
4. Efficient algebraic attacks with lower data complexity on certain stream ciphers.

Provable resistance of practical stream ciphers against algebraic attacks looks even harder than believed.

Questions ?