A Study on the Algebraic Immunity of Nonlinear Boolean Function in Cryptosystem

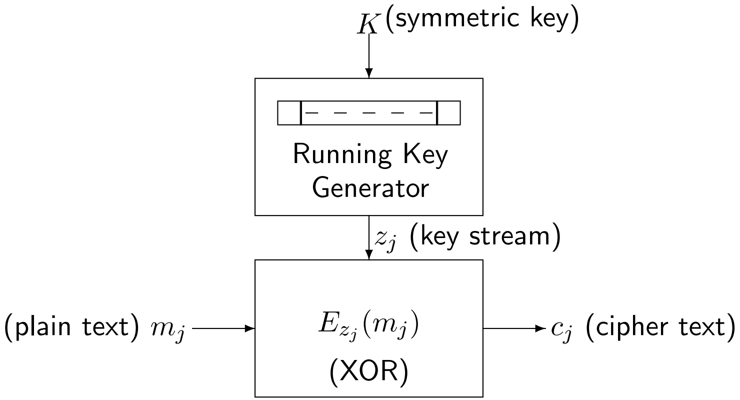
Ju Young Kim and Hong-Yeop Song

School of Electrical and Electronics Engineering,
Yonsei University, Seoul, Korea

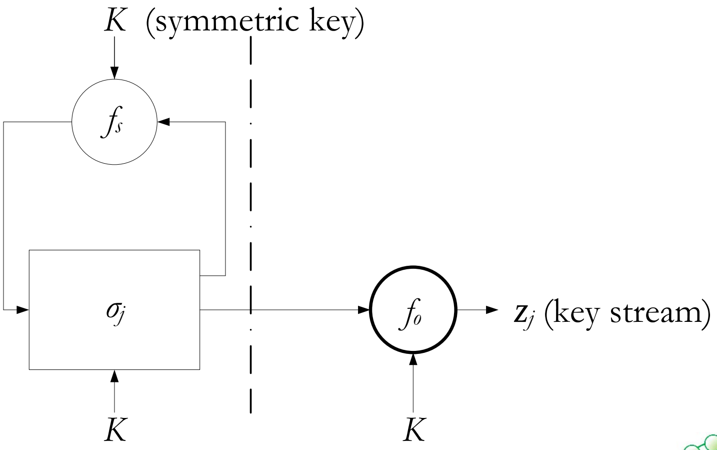
23, August, 2007

What we will discuss

Architecture of Stream Cipher



Architecture of Running Key Generator



f_o is described as (Nonlinear) Boolean Function

Boolean Function :

$$f_o : \mathbb{F}_2^n \to \mathbb{F}_2$$

ANF :

$$f(x_1, \cdots, x_n) = \bigoplus_{u \in \mathbb{F}_2^n} \lambda_u \left(\prod_{i=1}^{n} x_i^{u_i} \right), \lambda_u \in \mathbb{F}_2, u = (u_1, \cdots, u_n). \tag{1}$$

Algebraic Degree : $deg(f) \triangleq d$

- the maximal value of the Hamming weight of u such that $\lambda_u \neq 0$.
- In general $d = n - 1$.

Toy Example of f_o with $n = 4$

$$f(x_1, \cdots, x_4) = x_1 + x_2 + x_3 + x_4 + x_1x_3 + x_2x_3 + x_1x_2x_3 + x_2x_3x_4.$$

- $deg(f) = 3$
- equivalent to $(\mathbf{x}^{-1})\&1$
  - where $\mathbf{x}^{-1}$ is the map from $\mathcal{F}_{2^4}$ to $\mathcal{F}_2^4$
  - $\mathbf{x} \in \mathcal{F}_{2^4}$ is corresponding to $(x_1, \cdots, x_4) \in \mathcal{F}_2^4$
  - $\&$ is the operation to obtain specific(left most) component from the element of $\mathcal{F}_2^4$
- $f(x_1, \cdots, x_4)$ is balanced.

Algebraic Attack in General

- Given symmetric key of size N, and $d = deg(f)$ the number of equations to solve the system of simultaneous linear equations is [2003, Courtois]

$$\sum_{i=0}^{d} \binom{N}{i} \triangleq T(d)$$

- The computing complexity is $T(d)^{2.8}$[1969, Strassen]
- When $N = 80$ and $d = 15(n = 16)$, the complexity $\approx 2^{148}$
- If we **reduce the number of equations** needed, we can attack stream cipher easily. $\Rightarrow$ Reduce d.

Attack The Toy System($N = 80$)

$$f(x_1, \cdots, x_4) = x_1 + x_2 + x_3 + x_4 + x_1x_3 + x_2x_3 + x_1x_2x_3 + x_2x_3x_4.$$

$$\begin{array}{ll} \hline \mathbf{d} = 3 & \mathbf{T}(\mathbf{3}) = 2^{45} \\ \Downarrow & \\ \mathbf{d} = 2 & \mathbf{T}(\mathbf{2}) = 2^{33} \\ \hline \end{array}$$

- We can find two functions with degree 2
  - $g = x_1x_2 + x_1x_4$
  - $h = x_4x_3 + x_4x_2 + x_4x_1 + x_3x_2 + x_4$
- $g \cdot f = 0$ and $h \cdot f = h$
- $f = 1 \Rightarrow g = 0$ and $f = 0 \Rightarrow h = 0$
- g and h are annihilators of f

Algebraic Immunity

Annihilator $g(\neq 0)$ satisfies

$$f \cdot g \text{ or } (1+f) \cdot g \text{ vanishes.}$$

Algebraic Immunity: $AI(f)$

$$AI(f) = \min_{\substack{fg=0 \text{ or } (1+f)g=0 \\ g\neq 0}} \deg(g). \tag{2}$$

How to Calculate the AI of a Given Boolean Function

f : an n-variable boolean function of degree $> \lceil \frac{n}{2} \rceil \triangleq e$

$G = \{g_i | deg(g_i) \leq e, g_i \neq 0\}$

where, $1 \leq i \leq T(e)$,the number of equation needed.

$H = \{h_i | h_i = f \cdot g_i, g_i \in G\}$

g : defined as one of the linear combination of g_i's

has degree $\max_{g_i \in G} deg(g_i)$

$I = \{f \cdot g | g = \bigoplus_{i=1}^{T(e)} \lambda_i g_i, \forall \lambda = (\lambda_1, \cdots, \lambda_{T(e)}) \in \mathbb{F}_2^{T(e)}\}$

Then, the degree of the minimal degree member of the grӧbner basis of I is the $AI(f)$.[2003, Faugere]

Known Bounds

Theorem (Universal Bound, 2003, Courtois)

Let f be a Boolean function with n inputs. Then there is a Boolean function $g \neq 0$ of degree at most $\lceil n/2 \rceil$ such that fg is of degree at most $\lceil n/2 \rceil$.

Theorem (Bound for Boolean inverse function, 2006, Nawaz)

Let $f(x) = Tr_1^n(\beta x^{-1})$ and $g(x) = Tr_1^m(x^r)$. Then

$$deg(f(x)g(x)) = \lfloor \sqrt{n} \rfloor + \lceil \frac{n}{\lfloor \sqrt{n} \rfloor} \rceil - 2.$$

The Numbers of Equations to Attack ($N = 80$)

$\mathbf{n}$	$\mathbf{f}$				$\mathbf{f} = \mathbf{inv}^{(\mathbf{n})}$	
	$d = n - 1$	$\mathbf{T(d)}$	$\mathbf{d = AI(f)}$	$\mathbf{T(d)}$	$\mathbf{d = AI(f)}$	$\mathbf{T(d)}$
11	10	2^{41}	6	2^{29}	5	2^{25}
12	11	2^{44}	6	2^{29}	5	2^{25}
13	12	2^{47}	7	2^{32}	6	2^{29}
14	13	2^{50}	7	2^{32}	6	2^{29}
15	14	2^{53}	8	2^{35}	6	2^{29}
16	15	2^{56}	8	2^{35}	6	2^{29}
17	16	2^{59}	9	2^{38}	7	2^{32}
18	17	2^{62}	9	2^{38}	7	2^{32}
19	18	2^{65}	10	2^{41}	7	2^{32}

Design Approach

Find f_o with maximal AI.