Application of character sums to mathematical cryptography

-- Some examples in multivariate quadratic cryptography --

Tomohiro Harayama

Security Fundamentals Group
Information Security Research Center
National Institute of Information
and Communications Technology, JAPAN

These are the updated slides presented at JANT 17 (Tokyo University of Science) on 07/07/2007. If any comment or suggestion, please contact me (harayama@nict.go.jp).

Contents

- MQ problems (NP-complete) and Cryptosystems
- Weil sums and Weil sum algorithm
- Number of solutions of equations (Formula? or #P-complete)
- Weak Dembowski-Ostrom Polynomials (Potential existential forgery if used in Digital Signature)
- Summary (How can we do design-analysis-redesign ?)

Multivariate Quadratic (MQ) Problems and Cryptosystems

Oneway function

X, Z :sets. $f : X \to Z$ is called a ***oneway function*** if:

- It is "easy" to compute $z = f(x)$ for $\forall x \in X$

- For essentially all elements $z \in Im(f)$ the image by it is computationally infeasible to find any $x \in X$ such that $z = f(x)$

It is well known that:

Existence of $f : X \to Z \Rightarrow$ $\mathrm{P} \neq \mathcal{NP}$.

Cryptographic Reference Problems

- Integer factorization (e.g., RSA problem)
- Discrete logarithm (e.g., $\mathbb{Z}_p$, $\mathcal{EC}(\mathbb{F}_q)$)
- Lattice reduction (e.g., SVP problem)
- *Systems of multivariate quadratic polynomials over finite fields*
- Others

MQ (NP-complete problem)

$(y_1, \ldots, y_n) \in \mathbb{F}_2^n$ ← **Easy** / **Very hard** → $(x_1, \ldots, x_n) \in \mathbb{F}_2^n$

Hashed message — **Signature**

$$
\begin{cases}
y_1 = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(1)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(1)} x_i + \gamma_i^{(1)} \\
y_2 = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(2)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(2)} x_i + \gamma_i^{(2)} \\
\vdots \\
\vdots \\
y_n = \sum_{1 \le i \le j \le n} \alpha_{i,j}^{(n)} x_i x_j + \sum_{i=1}^{n} \beta_i^{(n)} x_i + \gamma_i^{(n)}
\end{cases}
$$

Birthday Attack on Hash Function

Let be $h : D \rightarrow R$ a hash function from domain D to R. Then, birthday attack is to find at least one collision $x, x' \in D$ s.t. $x \neq x'$ and $h(x) = h(x')$.

Note: a sorted list H
For i = 0 to |R|:
*** Select x uniformly at random with replacement from D.**
*** Compute y = h(x) and store pair (y, x) in H (sorted in y).**
*** If (some pair (y', x') in H) s.t.**
y' = y and x' != x,
then return (x, x', y), i // collision !
Return Null

Note: a sorted list H

For i = 0 to |R|:

*** Select x uniformly at random with replacement from D.**

*** Compute y = h(x) and store pair (y, x) in H (sorted in y).**

*** If (some pair (y', x') in H) s.t.**

y' = y and x' != x,

then return (x, x', y), i // collision !

Return Null

Expected number of steps (memory and time) is $O(\sqrt{|R|})$ because the number of ordered pairs (i_1, i_2) with $1 \leq i_1 \leq i_2 \leq i$ grows **quadratically** w.r.t. i (i.e., $\frac{i(i-1)}{2}$), so does probability for having collisions.

Birthday Attack on Digital Signature based on $MQ(p,n,n)$-Trapdoor $\mathbb{F}_p^n \to \mathbb{F}_p^n$

A ***Birthday Attack*** is to find at least one pair

$sig, sig' \in D$ s.t.

$$sig \neq sig' \text{ and}$$

$$F(sig) = F(sig') = message.$$

Note: a sorted list H.
For k = 0 to p^n:
 * Select sig uniformly at random
   with replacement from GF(p^n).
 * Compute msg = F(sig)
   store pair (msg, sig) in H.
 * If some (msg', sig') s.t.
   msg = msg' and sig' != sig return (sig, sig', msg) k;
Return Null.

Is it really $O(\sqrt{2^n})$ for all $F \in \mathcal{MQ}(2^n, n, n)$?

Kipnis-Shamir's Lemma

Let *standard linear bijection* $\phi : \mathbb{F}_{q^n} \to \mathbb{F}_q^n$.

For $\forall$ multi-polys $(P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ over $\mathbb{F}_q$,

$\exists$ uni-poly $f \in \mathbb{F}_{q^n}[x]$:

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i}+q^{\beta_i}} + \sum_{j=1}^{L} b_j x^{q^{\gamma_j}} + c,$$

such that: $\forall (x_1, \ldots, x_n) \in F_q^n$

$$\phi \circ f \circ \phi^{-1}(v_1, \ldots, v_n) = (P_1(v_1, \ldots, v_n), \ldots, P_n(v_1, \ldots, v_n))$$

Central Polynomial

Given $(P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ of $MQ(q, n, n)$. A ***central polynomial*** over $\mathbb{F}_q$: is of form:

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i}+q^{\beta_i}} + \sum_{j=1}^{L} b_j x^{q^{\gamma_j}} + c.$$

$$F : \mathbb{F}_q^n \to \mathbb{F}_q^n \overset{Kipnis-Shamir}{\Leftrightarrow} f : \mathbb{F}_{q^n} \to \mathbb{F}_{q^n} \text{ (central)}$$

DO Polynomial

Homogeneous $(P_1(x_1, \ldots, x_n), \ldots, P_n(x_1, \ldots, x_n))$ of $MQ(q, n, n)$. A ***DO polynomial*** over $\mathbb{F}_q$: is of form:

$$f(x) = \sum_{i=1}^{D} a_i x^{q^{\alpha_i}+q^{\beta_i}}$$

$F : \mathbb{F}_q^n \to \mathbb{F}_q^n \overset{Kipnis-Shamir}{\Leftrightarrow} f : \mathbb{F}_{q^n} \to \mathbb{F}_{q^n}$ DO **Homogeneous**

Weil Sum and Weil Sum Algorithm

Simplified Central Polynomial

$f(x)$: ***central polynomial*** over $\mathbb{F}_q$:
$(f(0) = 0,\ 1 \leq i \leq D)$ s.t.

$$t_i, y_i, s_i \in \mathbb{Z},$$
$$A_i \text{ s.t. } A_i^{p^{t_i}} = a_i \in \mathbb{F}_q,$$
$$b = \sum_{j=1}^{L} b_j^{p^{e-\gamma_j}}$$
$$\beta_i \in \mathbb{F}_q \text{ s.t. } t_i \equiv \beta_i - \beta_1 \mod n,$$
$$y_i = n - s_i \ (2 \leq i \leq D),$$
$$s_i = \alpha_i - \beta_i \geq 0.$$

Simplified Central Polynomial

Weil sum values are **equivalent** for general and simplified central polynomials

$$f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x$$

so that we work on the Weil sum:

$$S(a_1, \ldots, a_D, b_1, \ldots, b_L)$$
$$= \sum_{x \in \mathbb{F}_q} \chi_1\left(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x\right)$$

Auxiliary Linearized Polynomial

The computational complexity of Weil sum algorithm is governed by the **dimension**.

The product of Weil sum $|S|$ is

$$|S|^2 = q \sum_{T_D(w)=0, w\in\mathbb{F}_q} \chi_1\Big(\sum_{i=1}^{D} A_i w^{p^{s_i}+1} + b^{p^{\beta_1}} w\Big).$$

The index w of the outer sum runs throughout the **set of roots** in $\mathbb{F}_q$ of a linearized polynomial:

$$T_D(w) = A_1^{p^{s1}} w^{p^{2s_1}} + A_1 w + \sum_{i=2}^{D} \big[A_i^{p^{s1}} w^{p^{s_1+s_i}} + (A_i w)^{p^{s_1+y_i}}\big].$$

Root of Auxiliary Linearized Polynomial

$$T_D(w) = A_1^{p^{s1}} w^{p^{2s_1}} + A_1 w + \sum_{i=2}^{D} [A_i^{p^{s1}} w^{p^{s_1+s_i}} + (A_i w)^{p^{s_1+y_i}}].$$

$$\varepsilon = \gcd_{2 \le i \le D}(2s_1, s_1 + s_i, s_1 + y_i, n).$$

The **set of roots** is:

- a linear subspace $\subset \mathbb{F}_q$ over $\mathbb{F}_{p^\varepsilon}$
- $\cong \mathbb{F}_{p^{t\varepsilon}}$ for some integer $t \in \mathbb{Z}$.

The dimension $t\epsilon$ of the roots of $T_D(w)$ is defined by portions of exponents $2s_1, s_1 + s_i, s_1 + y_i$ and n.

Character ($p = 2$)

- Let $\mathbb{F}_q$ be of characteristic $p = 2$. Then, for any $u \in \mathbb{F}_q$, $\chi_1(u) = \exp(2\pi i Tr(u)/2)$ is real.

- Let $p = 2$ and

$$S = \sum_{x \in \mathbb{F}_q} \chi_1\left(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x\right)$$:

the Weil sum of central polynomial. Then, S is real and $|S|^2 = S^2$.

Summing 1 or -1 for at most q times.

Weil Sum Algorithm ($p = 2$)

INPUT $f(x) = \sum_{i=1}^{D} a_i x^{2^{\alpha_i}+2^{\beta_i}} + \sum_{i}^{L} b_i x^{2^{\gamma_i}}$: central polynomial in $\mathbb{F}_{2^n}[x]$.
OUTPUT $|S|$: the absolute value of Weil sum S of $f(x)$.

---

1. Compute $T_D(x) \in \mathbb{F}_{2^n}[x]$ (Suppose the rank of the kernel is l).
2. Compute the basis $\{\gamma_1, \ldots, \gamma_l\}$ of $ker(T_D)$.
3. Let U be $0 \in Z$.
4. Compute $\gamma_{i,j_1,j_2} = Tr(A_i \gamma_{j_1} \gamma_{j_2}^{2^{s_i}})$ for $1 \leq i \leq D, 1 \leq j_1, j_2 \leq l$
5. Compute $\rho_j = Tr(b^{2^{\beta_1}} \gamma_j)$ for $1 \leq j \leq l$.
6. For each $(x_1, \ldots, x_l) \in \mathbb{F}_2^l$, evaluate:

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} + \sum_{j=1}^{l} x_j \rho_j \in \mathbb{F}_2$$

   and set $U = U + C_{(x_1,\ldots,x_l)}$. (Note: **integer addition**.)
7. Return $2^{n/2}\sqrt{2^l - 2U}$. (Note: **absolute value**)

1. Compute $T_D(x) \in \mathbb{F}_{2^n}[x]$ (Suppose the rank of the kernel is l).
2. Compute the basis $\{\gamma_1, \ldots, \gamma_l\}$ of $ker(T_D)$.
3. Let U be $0 \in Z$.
4. Compute $\gamma_{i,j_1,j_2} = Tr(A_i \gamma_{j_1} \gamma_{j_2}^{2^{s_i}})$ for $1 \leq i \leq D, 1 \leq j_1, j_2 \leq l$
5. Compute $\rho_j = Tr(b^{2^{\beta_1}} \gamma_j)$ for $1 \leq j \leq l$.
6. For each $(x_1, \ldots, x_l) \in \mathbb{F}_2^l$, evaluate:

Go through each element in kernel

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} + \sum_{j=1}^{l} x_j \rho_j \in \mathbb{F}_2$$

   and set $U = U + C_{(x_1,\ldots,x_l)}$. (Note: **integer addition**.)
7. Return $2^{n/2}\sqrt{2^l - 2U}$. (Note: **absolute value**)

Time complexity: $O(C_D L l^2 (n^3 + 2^l))$

Number of Solutions of Equations

Batched Solutions

Set $\delta = \gcd(s_1, \ldots, s_D, n)$. Let $f(x)$ be a simplified ***Dembowski-Ostrom polynomial*** $\sum_{i=1}^{D} A_i x^{p^{s_i}+1}$ over with $\mathbb{F}_q$ each s_i/δ odd and $g(x,y) = f(x) - y^{p^\delta} + y$ the bivariate polynomial. Then, the number of solutions $N(g(x,y))$ of the bivariate equation $g(x,y) = 0$ is estimated as:

$$N \equiv -1 \mod p^\delta + 1.$$

Condition s_i/δ: odd

$\Longrightarrow$ congruential estimate of $N(f(x) - y^{p^\delta} + y)$

GCD and Exponents

$n, s_i\ (1 \le i \le D)$: non-negative integers with

$$\delta = \gcd(s_1, \ldots, s_D, n).$$

s_i/δ : odd for each $1 \le i \le D$ forces s_i to be positive.

Assume that n/δ is even. Then we have:

$$\left(\frac{2^{s_i}+1}{2^{\delta}+1}, 2^{\delta} - 1\right) = 1.$$

Proof (idea):

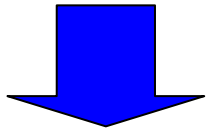
$$2^{s_i} + 1 = (2^{\delta} + 1)(2^{(s_i/\delta - 1)\delta} - 2^{(s_i/\delta - 2)\delta} + \ldots - 2^{\delta} + 1).$$

even odd …….. odd, even

Bivariate Equation. Emulation Condition

Let $f(x)$ a DO poly $\sum_{i=1}^{D} A_i x^{p^{s_i}+1} over \mathrm{F}_q$. $\delta = (s_1, \ldots, s_D)$.

n/δ is even,
$\delta = (s_i, e)$ for each i,
s_i/δ is odd for each i, and
2δ divides $s_i - s_j$ for all $i \neq j$.



$$N(f(x,y)) = q + (p^{\delta} - 1)S.$$

Relating # solutions N to Weil sum

$$S = \sum_{x \in \mathbb{F}_q} \chi_1\left(\sum_{i=1}^{D} A_i x^{p^{s_i}+1} + b^{p^{\beta_1}} x\right)$$

Simplified Weil Sum Algorithm (p=2)

INPUT $f(x) = \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$: DO poly with EC.
OUTPUT S: Weil sum of $f(x)$

---

1. Compute $T_D(x) \in \mathbb{F}_{2^n}[x]$ (Suppose the rank of the kernel is l).
2. Compute the basis $\{\gamma_1, \ldots, \gamma_l\}$ of $ker(T_D)$.
3. Let U be $0 \in Z$.
4. Compute $\gamma_{i,j_1,j_2} = Tr(A_i \gamma_{j_1} \gamma_{j_2}^{2^{s_i}})$ for $1 \le i \le D, 1 \le j_1, j_2 \le l$
5. For each $(x_1, \ldots, x_l) \in \mathbb{F}_2^l$, evaluate:

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} \in \mathbb{F}_2$$

   and set $U = U + C_{(x_1,\ldots,x_l)}$. (Note: **integer addition**.)

6. If $|S| = 2^{n/2}\sqrt{2^l - 2U}$ satisfies $2(1 - |S|\ |) \equiv 0 \mod 2^{\delta} + 1$ return $|S|$. Otherwise return $-|S|$.

1. Compute $T_D(x) \in \mathbb{F}_{2^n}[x]$ (Suppose the rank of the kernel is l).
2. Compute the basis $\{\gamma_1, \ldots, \gamma_l\}$ of $ker(T_D)$.
3. Let U be $0 \in Z$.
4. Compute $\gamma_{i,j_1,j_2} = Tr(A_i \gamma_{j_1} \gamma_{j_2}^{2^{s_i}})$ for $1 \leq i \leq D, 1 \leq j_1, j_2 \leq l$
5. For each $(x_1, \ldots, x_l) \in \mathbb{F}_2^l$, evaluate:

$$C_{(x_1,\ldots,x_l)} = \sum_{i=1}^{D} \sum_{j_1=1}^{l} \sum_{j_2=1}^{l} x_{j_1} x_{j_2} \gamma_{i,j_1,j_2} \in \mathbb{F}_2$$

and set $U = U + C_{(x_1,\ldots,x_l)}$. (Note: **integer addition**.)

Go through each element in kernel

6. If $|S| = 2^{n/2}\sqrt{2^l - 2U}$ satisfies $2(1 - |S|\,|) \equiv 0 \mod 2^{\delta} + 1$ return $|S|$. Otherwise return $-|S|$.

Time complexity: $O(D\, l^2 (n^3 + 2^l))$

Large # of Solutions: Potential cryptanalytic implications

$$S = \{(x, y) \in \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} \mid \sum_{i=1}^{D} A_i x^{2^{s_i}+1} = y^{2^\delta} - y\}$$

If **S is large**, we pick up random $(x, y) \in \mathbb{F}_{2^n} \times \mathbb{F}_{2^n}$ and the probability that

$x \to \sum_{i=1}^{D} A_i x^{2^{s_i}+1}$ **=?** $y^{2^\delta} - y \leftarrow y$

is higher.

Oneway: Hard to invert.

Meet-in-the-Middle?

Linear: easy to Invert ?

Weak Dembowski-Ostrom Polynomials

Existential Forgery (if Signature Scheme)

An adversary is able to forge a signature of **least one** message over which the adversary has little or no control.

Suppose we consider a key subspace of:

$$\mathcal{K} = \{ \sum_{i=1}^{D} A_i x^{2^{s_i}+1} \}$$

Ideally, **none** of key in $\mathcal{K}$ should be weak.

Linearized Binomial Attack

Let $f(x)$ be a secret Dembowski-Ostrom polynomial:

$$f(x) = \sum_{i=1}^{D} A_i x^{p^{s_i}+1} = A_1 x^{p^{s_1}+1} + \cdots + A_D x^{p^{s_D}+1},$$

where $D \geq 1$, $A_i \in \mathbb{F}_q$ for $1 \leq i \leq D$ and

$$0 \leq s_1 < s_2 < \ldots < s_D \leq q-1.$$

With $D \geq 1$ and $\delta = (s_1, \ldots, s_D, n)$ such that:

$$\mathcal{K} = \{\sum_{i=1}^{D} A_i x^{2^{s_i}+1}, n/\delta \text{ even}, \delta = (s_i, n), s_i/\delta \text{ odd}, 2\delta \text{ divides } s_i - s_j\}.$$

$\mathcal{K}$ is the key space of our attack.

A *linearized binomial attacker* against $MQ(2,n,n)$-trapdoor

$$F(x) = (P_1(x_1,\ldots,x_n),\ldots,P_n(x_1,\ldots,x_n)).$$

Randomly guess the value of the unknown $\delta = (s_1,\ldots,s_D,n)$ from $\{1,\ldots,n\}$. This δ allows the adversary to fix a linearized binomial $L(y) = y^2 - y$ in $\mathbb{F}_q[y]$.

We denote by $Im(L)$ the image of the mapping L over $\mathbb{F}_q$.

1. Generate $\frac{T}{t}2^{\frac{n-\delta}{2}}$ random elements $x \in \mathbb{F}_q$ and obtain the list:

$$\{f(x_1), f(x_2), \ldots, f(x_{\frac{T}{t}2^{\frac{n-\delta}{2}}})\}.$$

2. Generate $2^{\frac{n-\delta}{2}}$ random elements $z \in Im(L)$ to obtain the list:

$$\{z_1,\ldots,z_{2^{\frac{n-\delta}{2}}}\}.$$

3. Search for a coincidence $f(x_j) = z_i$ for some i,j in the two lists.

Complexity: linearized binomial attacker must generate

$$\frac{T}{t}2^{\frac{n-\delta}{2}} > 2^{\frac{n-\delta}{2}}$$

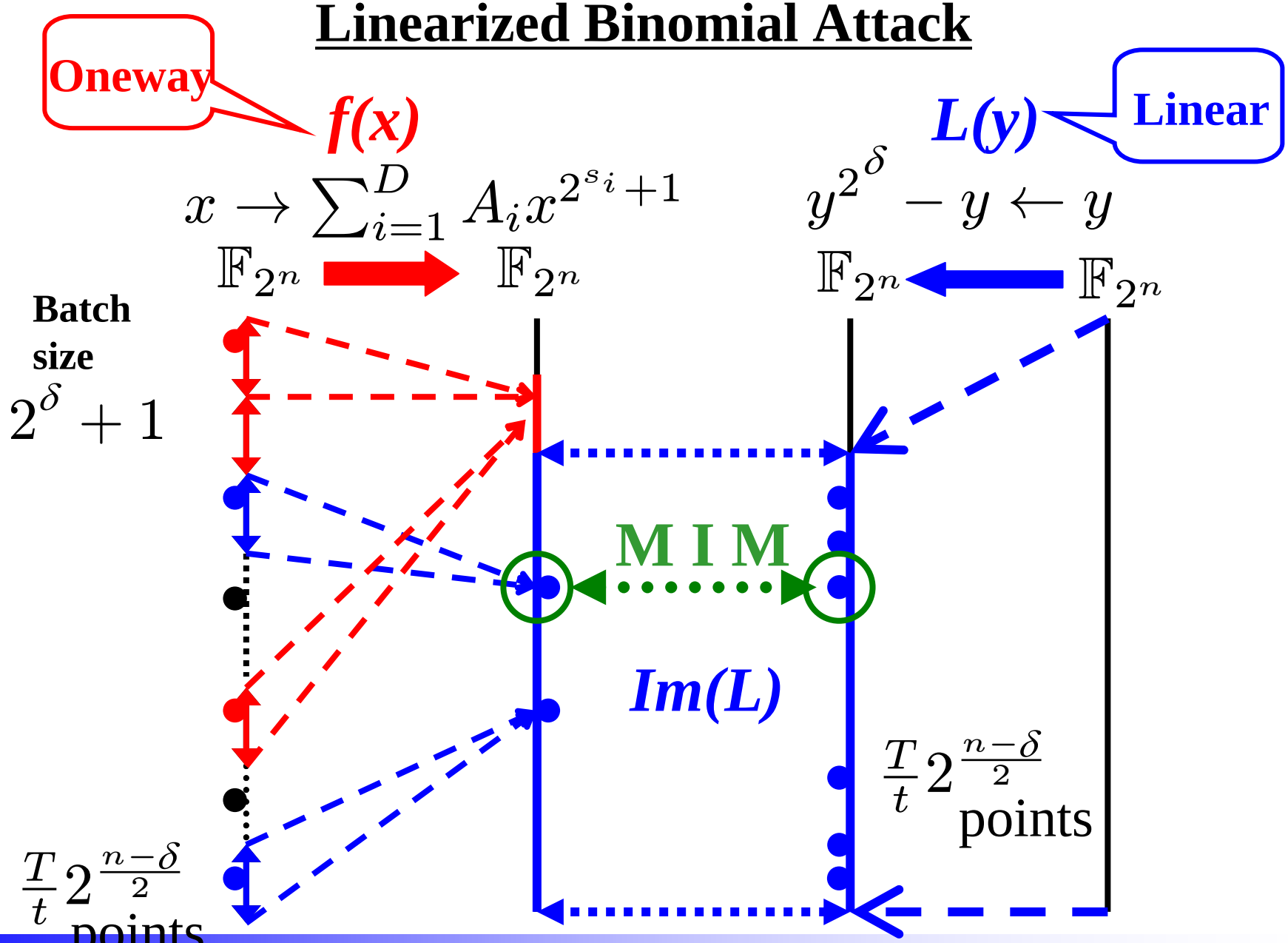
elements x's for their images $f(x)$'s in order to obtain the list

$$\{f(x_1), f(x_2), \ldots, f(x_{\frac{T}{t}2^{\frac{n-\delta}{2}}})\}$$

in which at least $2^{n/\delta}$ elements are expected to be inside $Im(L)$.
δ is guessed from $\{1, \ldots, n\}$.

The total time complexity: $O(n \times \frac{T}{t}2^{\frac{n-\delta}{2}})$.

Linearized Binomial Attack



n/4

$$f(x) = A_1 x^{2^{s_1}+1} + A_2 x^{2^{s_2}+1} \text{ (i.e. } D = 2)$$

For every integer $i \geq 1$, define the even extension degree n of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$ and (s_1, s_2): exponents of $f(x)$ such as:

$$\begin{cases} n = 4i, \\ s_1 = i, \\ s_2 = 3i. \end{cases}$$

where $\begin{cases} \delta = (s_1, n) = (s_2, n) = i, \\ n/\delta = 4 : \text{ even }, \\ s_1/\delta = 1 \text{ odd }, s_2/\delta = 3 \text{ odd and} \\ 2\delta = 2i \text{ divides } |s_2 - s_1|. \end{cases}$

Thus, we have $\delta = n/4$.

Weak DO Polynomials (*D=2, delta = n/4*)

$f(x) = x^{2^{s1}+1} + x^{2^{s2}+1}$: DO-poly in $F(2^n)[x]$ (n from 8 to 24)

$n = 8, (s_1, s_2) = (2, 6)$:

$$f(x) = x^{2^2+1} + x^{2^6+1} = x^5 + x^{65} \text{ in } \mathbb{F}_{2^8}[x].$$

$n = 12, (s_1, s_2) = (3, 9)$:

$$f(x) = x^{2^3+1} + x^{2^9+1} = x^9 + x^{513} \text{ in } \mathbb{F}_{2^8}[x].$$

$n = 16, (s_1, s_2) = (4, 12)$:

$$f(x) = x^{2^4+1} + x^{2^{12}+1} = x^{17} + x^{4097} \text{ in } \mathbb{F}_{2^{16}}[x].$$

$n = 20, (s_1, s_2) = (5, 15)$:

$$f(x) = x^{2^5+1} + x^{2^{15}+1} = x^{33} + x^{32769} \text{ in } \mathbb{F}_{2^{20}}[x].$$

$n = 24, (s_1, s_2) = (6, 18)$:

$$f(x) = x^{2^6+1} + x^{2^{18}+1} = x^{65} + x^{262145} \text{ in } \mathbb{F}_{2^{24}}[x].$$

Summary (How can we do design-analysis-redesign ?)

1. Design: Given a cryptosystem as it is.
2. Analysis
    1. Any formula ? (e.g., # of solutions of polynomial equations)
    2. Any algorithm ? (e.g., Weil sum evaluation algorithm)
    3. Any relations ? (e.g., Weil sum value to the number of solutions)
    4. Any attack? (e.g., Attack algorithm to characterize “weak” keys)
    5. Any experiment ? (e.g., Proof of existence of such weak keys. If exists on smaller parameters (e.g. n), try to derive some formula for n of any size)
3. (Try to) Redesign it (e.g., Think how to eliminate weak key class from trapdoor structure or key generation algorithm).

References

- Aviad Kipnis and Adi Shamir, Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization, CRYPTO '99, LNCS 1666, pp. 19-30, 1999.
- Robert S. Coulter, Explicit Evaluations of Some Weil Sums, Acta Arithmetica, Vol. 83, pp. 241-251, 1998.
- Robert S. Coulter, Further Evaluations of Weil Sums, Acta Arithmetica, Vol. 86, pp. 217-226, 1998.
- Robert S. Coulter, On the Evaluation of a Class of Weil Sums in Characteristic 2, NZ J. Mathematics, Vol. 28, No. 2, pp. 171-184, 1999.
- Donald Mills, On the Evaluation of Weil Sums of Dembowski-Ostrom Polynomials, Journal of Number Theory, Vol. 92, No. 1, pp. 87-98, 2002.
- Tomohiro Harayama and Donald K. Friesen, Weil Sum for Birthday Attack in Multivariate Quadratic Cryptosystem, Journal of Mathematical Cryptology, Vol. 1, No. 1, pp.79-104, 2007.