Algebraic Attacks on RAIN and AIM Using Equivalent Representations

Fukang Liu[1], Mairon Mahzoun[2], Morten Øygarden[3], Willi Meier[4]

[1] Tokyo Institute of Technology, Tokyo, Japan
[2] Eindhoven University of Technology
[3] Simula UiB, Bergen, Norway
[4] FHNW, Windisch, Switzerland

28th March 2024

Introduction

AIMer and Rainier

Post-quantum secure digital signature schemes based on the MPC-in-the-Head paradigm.

AIMer

- Round 1 candidate for NIST PQC.
- One of the 4 round 2 candidates for KpqC.
- Security relies on AIM (CCS 2023).

Rainier

- Shorter Signatures Based on Tailor-Made Minimalist Symmetric-Key Crypto.
- Security relies on RAIN (CCS 2022).

Security Analysis

Analyze the security of AIM and RAIN against algebraic attacks.

Multivariate Polynomial Equations

Model the cryptographic primitive as a set of polynomials:

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ \qquad \vdots \\ f_m(x_1, \ldots, x_n) = 0 \end{cases} \qquad deg(f_i) = d_i, i \in \mathbb{K}[x_1, \ldots, x_n]$$

Find the set of solutions:

$$V(f_1, \ldots, f_m) = \left\{(x_1, \ldots, x_n) \in \overline{\mathbb{K}}^n : f_i(x_1, \ldots, x_n) = 0, \forall i \in [1, m]\right\}$$

Finding solutions

- Fast Exhaustive Search.
- Crossbred Algorithm.
- Polynomial Method.

Fast Exhaustive Search

Exhaustive Search

Idea: Compute $f_i(x_1, \ldots, x_n)$ for all $(x_1, \ldots, x_n) \in \{0,1\}^n$.

$$2^n \cdot \sum_{j=0}^{deg(f_i)} \binom{n}{j},$$

for each f_i.

Fast Exhaustive Search

Idea: Compute $f_i(x_1, \ldots, x_n)$ for all $(x_1, \ldots, x_n) \in \{0,1\}^n$.

$$deg(f_i) \cdot 2^n.$$

Time Complexity(bits)	Memory Complexity
$4d \cdot \log_2 n \cdot 2^n$	$n^2 \cdot \sum_{j=0}^{d} \binom{n}{j}$

Crossbred Algorithm

Macaulay Matrix:

$$\mathcal{F} = \begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ \vdots \\ f_m(x_1, \ldots, x_n) = 0 \end{cases}$$

$$\mathrm{Mac}_D(\mathcal{F}) = \begin{array}{c} \\ \vdots \\ uf_i \\ \vdots \end{array} \begin{array}{c} \ldots \quad x_i \quad \ldots \\ \begin{pmatrix} & & \\ & c_i & \\ & & \end{pmatrix} \end{array}$$

$deg(uf_i) \leq D$
c_i: coefficient of x_i in uf_i

Crossbred Idea

Guess k variables, and derive an easy system of degree $d' < D$ to solve.

Polynomial Method

Find the solutions to a smaller number of equations and check the solutions.

- Accurate time complexity.

Time Complexity(bits)	Memory Complexity
$n^2 \cdot 2^{0.815n}$	$n^2 \cdot 2^{0.63n}$

- No gain if the system is over-defined.
  - More polynomials in the system, more information about the solution.

We analyze the security of AIM, and Rain using the approaches described.

Rain

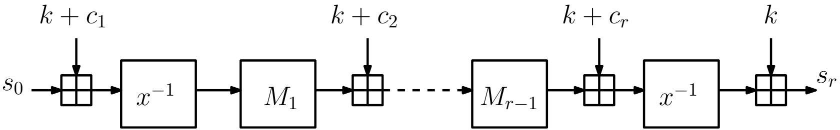


Figure: The r-round Rain: Rain_r.

$$M_i(x) = \sum_{j=0}^{n-1} a_{i,j} x^{2^j}$$

$M_i(x)$:

High degree polynomial over $\mathbb{F}_{2^n}$, linear over $\mathbb{F}_2$.

Attack Goal

- Recover k from (s_0, s_1) with $O(1)$ data complexity.
- We are interested in Rain_2 and Rain_3.

Rain$_2$: Low Degree Representation

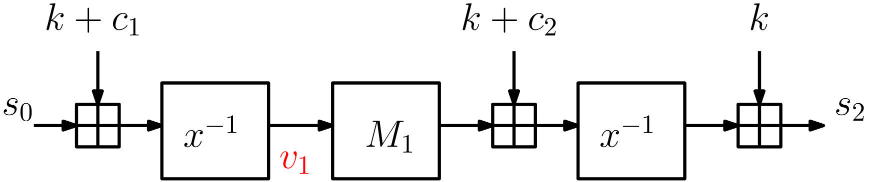


Figure: Rain$_2$

Derive n boolean equations in n variables:

$$v_1 M_1(v_1) + t_1 {v_1}^2 M_1(v_1) = 1 + t_1 v_1 + t_0 v_1 + t_0 t_1 {v_1}^2 + {v_1}^2$$
$$t_0 = s_0 + c_1 + c_2, \quad t_1 = s_0 + c_1 + s_2.$$

n	Time Complexity(bits)	Memory Complexity
128	2^{118}	2^{95}
192	2^{172}	2^{136}
256	2^{225}	2^{177}

Rain_2: Low-memory Analysis

The polynomial system describing Rain_2 is:

$$F(v_1) = v_1 M_1(v_1) + t_1 {v_1}^2 M_1(v_1) = 1 + t_1 v_1 + t_0 v_1 + t_0 t_1 {v_1}^2 + {v_1}^2 \quad (1)$$

We derive:

$$G(v_1) = M_1(v_1) F(v_1) \quad (2)$$

$$H(v_1) = (v1 + t_1 {v_1}^2) F(v_1) \quad (3)$$

- (1)-(3) form a quadratic polynomial system with $3n$ equations in n variables.
- Solve using Crossbred algorithm.

Rain_2: Low-memory Analysis

Complexity of Crossbred

- Polynomials F, G, H are related, and have structure.
- In a higher degree, syzygies can appear.

Rank of degree D Macaulay matrix:

$$\text{Rank}(\mathcal{M}_{\leq D}(\mathcal{F})) \begin{cases} 3n & D = 2, \\ n(3n-8) + \text{Rank}(\mathcal{M}_{\leq 2}(\mathcal{F})) & D = 3, \\ 3n\binom{n}{2} - \binom{3n+1}{2} - 8n^2 + 17n + \text{Rank}(\mathcal{M}_{\leq 3}(\mathcal{F})) & D = 4. \end{cases}$$

Number of degree $\leq D$-monomials in n variables that have degree ≥ 2 in the first $i < n$ variables:

$$\text{Mon}_{n,D}(i) = \begin{cases} \binom{i}{2} & D = 2, \\ \binom{i}{3} + (n-i)\binom{i}{2} + \text{Mon}_{n,2}(i) & D = 3, \\ \binom{i}{4} + (n-i)\binom{i}{3} + \binom{n-i}{2}\binom{i}{2} + \text{Mon}_{n,3}(i) & D = 4. \end{cases}$$

Rain_2: Low-memory Analysis

- Find k and eliminate $\text{Mon}_{n,D}(k)$ monomials with Gaussian elimination.
- Guess $n-k$ variables.
- Solve linear system in the first k variables.

$$k = k(\mathcal{F}) = \max\left\{ i \in \mathbb{Z}_{>0} \;\middle|\; \text{Rank}(M_{\leq D}(\mathcal{F})) - \text{Mon}_{n,D}(i) \geq i \right\}.$$

Rain$_2$: Low-memory Analysis

Table: Cost analysis of various methods for solving

Method	n	k	Time (bits)	Memory (bits)
Polynomial Method	128	-	2^{118}	2^{95}
	192	-	2^{172}	2^{136}
	256	-	2^{225}	2^{177}
Crossbred $D = 2$	128	27	2^{115}	2^{22}
	192	33	2^{174}	2^{23}
	256	38	2^{234}	2^{25}
Crossbred $D = 3$	128	30	2^{113}	2^{35}
	192	36	2^{172}	2^{37}
	256	41	2^{231}	2^{40}
Crossbred $D = 4$	128	32	2^{111}	2^{45}
	192	38	2^{170}	2^{50}
	256	44	2^{228}	2^{52}

Rain$_3$

Can the same approach be used to attack more rounds?

$$\left(M_1(v_1) + \frac{1}{v_1} + t_0\right)\left(\frac{1}{v_1} + s_0 + c_1 + c_3 + M_2^{-1}\left(\frac{1}{\frac{1}{v_1} + s_0 + c_1 + s_3}\right)\right) = 1$$

Let

$$t_2 = s_0 + c_1 + c_3, \quad t_3 = s_0 + c_1 + s_3.$$

Then, we have

$$\left(v_1 M_1(v_1) + 1 + t_0 v_1\right)\left(1 + t_2 v_1 + v_1 M_2^{-1}\left(\frac{v_1}{1 + t_3 v_1}\right)\right) = v_1^2.$$

- $M_2^{-1}\left(\frac{v_1}{1+t_3 v_1}\right)$ has large algebraic degree.
- If either of $M_2^{-1}(x)$ or $M_1(x)$ are sparse, attack will be successful.

AIM

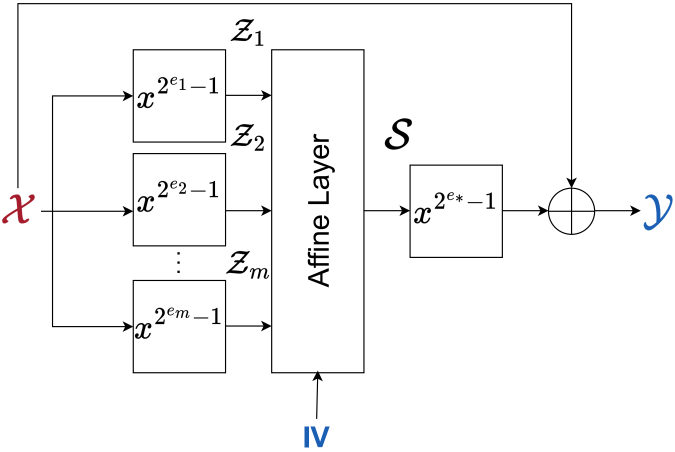


Figure: The AIM one-way function.

AIM

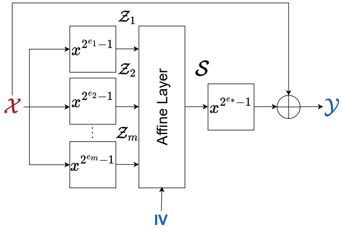
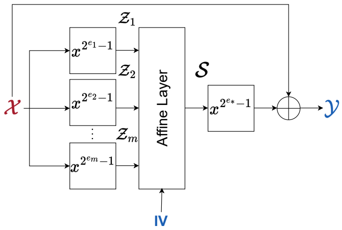


Figure: The AIM one-way function.

Scheme	n	Field	m	e_1	e_2	e_3	e_*
AIM-I	128	$\mathbb{F}_{2^{128}}$	2	3	27		5
AIM-III	192	$\mathbb{F}_{2^{192}}$	2	5	29		7
AIM-V	256	$\mathbb{F}_{2^{256}}$	3	3	53	7	5

Table: Instances of AIM for different security levels.

Fast Exhaustive Search



$$\mathcal{X} = \mathcal{S}^{2^{e_*}-1} + \mathcal{Y} \tag{4}$$

$$\mathcal{Z}_i = \left(\mathcal{S}^{2^{e_*}-1} + \mathcal{Y}\right)^{2^{e_i}-1} \tag{5}$$

$$\mathcal{Z}_i = \sum_{j=0}^{2^{e_i-1}} \mathcal{Y}^j \mathcal{S}^{2^{e_*-1}(2^{e_j}-1-j)} \tag{6}$$

Fast Exhaustive Search

$$\mathcal{Z}_m = B_m^{-1}\left(c + \mathcal{S} + \sum_{i=1}^{m-1} B_i\left(\left(\mathcal{S}^{2^{e_*}-1} + \mathcal{Y}\right)^{2^{e_i}-1}\right)\right) \quad (7)$$

$$\mathcal{Z}_m = \left(\mathcal{S}^{2^{e_*}-1} + \mathcal{Y}\right)^{2^{e_m}-1} \quad (8)$$

with algebraic degree of d_{max}

$$B_{m-1}^{-1}\left(c + \mathcal{S} + \sum_{i=1}^{m-1} B_i\left((\mathcal{S}^{2^{e_*}-1} + \mathcal{Y})^{2^{e_i}-1}\right)\right)(\mathcal{S}^{2^{e_*}-1} + \mathcal{Y}) = (\mathcal{S}^{2^{e_*}-1} + \mathcal{Y})^{2^{e_m}} \quad (9)$$

n boolean equations of algebraic degree upper bounded by $d_{max} + e_m$.

AIM

Scheme	n	$m+1$	Algebraic Degree	Time	Memory	Complexity
AIM-I	128	3	10	$2^{136.2}$	$2^{61.7}$	2^{115}
AIM-III	192	3	14	$2^{200.7}$	$2^{84.3}$	2^{178}
AIM-V	256	4	15	$2^{265.0}$	$2^{95.1}$	2^{241}

Table: Summary of results for AIM

Conclusion

Smart ways to model a cryptographic primitive $\Rightarrow$ Lower complexity to recover the secrets.

Rain_2

Using the Polynomial method, and Crossbred, all instances are broken.

Rain_3

If the linear layer is sparse, it is not secure.

AIM

Using Fast Exhaustive Search, all instances of AIM are broken.

The end

Thank you for your attention![1]

[1] Photo of Alpine Choughs in Italian Alps.