New attacks on Biscuit signature scheme

Charles Bouillaguet, Julia SAUVAGE

Almasty, lip6

October 17, 2023

Biscuit

Biscuit signature scheme [Bettale et al., 23]

- Round-1 submission to the NIST competition for additional post-quantum signatures
- `MPC-in-the-Head`-based Signature.
- m structured algebraic equations in n variables ($m \approx n$) over $\mathbb{F}_q$.
- With $\mathbf{x} = \{x_1, \ldots, x_n\} \in \mathbb{F}_q^n$, u_i, v_i and w_i affine forms :

$$p_i(\mathbf{x}) = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x}) \tag{1}$$

$i \in \{1, \ldots, m\}$

Biscuit

Biscuit signature scheme [Bettale et al., 23]

- Round-1 submission to the NIST competition for additional post-quantum signatures
- `MPC-in-the-Head`-based Signature.
- m structured algebraic equations in n variables ($m \approx n$) over $\mathbb{F}_q$.
- With $\mathbf{x} = \{x_1, \ldots, x_n\} \in \mathbb{F}_q^n$, u_i, v_i and w_i affine forms :

$$p_i(\mathbf{x}) = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x}) \tag{1}$$

$i \in \{1, \ldots, m\}$

Attack complexity

- Combinatory algo : $q^{\frac{3}{4}n}$.
- asymptotic complexity Hybrid Method : $2^{2.01n}$

Biscuit

Biscuit signature scheme [Bettale et al., 23]

- Round-1 submission to the NIST competition for additional post-quantum signatures
- `MPC-in-the-Head`-based Signature.
- m structured algebraic equations in n variables ($m \approx n$) over $\mathbb{F}_q$.
- With $\mathbf{x} = \{x_1, \ldots, x_n\} \in \mathbb{F}_q^n$, u_i, v_i and w_i affine forms :

$$p_i(\mathbf{x}) = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x}) \tag{1}$$

$i \in \{1, \ldots, m\}$

Attack complexity

- Combinatory algo : $q^{\frac{3}{4}n}$.
- asymptotic complexity Hybrid Method : $2^{2.01n}$

Our new algorithm

- direct : $n^3 q^{\frac{n}{2}}$.
- New hybrid approach : $2^{1.59n}$

New idea

We have

$$p_i(\mathbf{x}) = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x}) \tag{2}$$

We guess $v_i(\mathbf{x}) = a \in \mathbb{F}_q$. We have now:

$$\begin{aligned} p_i(\mathbf{x}) &= u_i(\mathbf{x}) + a \times w_i(\mathbf{x}) \\ v_i(\mathbf{x}) &= a \end{aligned}$$

$\hookrightarrow$ $m - 1$ polynomials in $n - 2$ variables.

New idea

We have

$$p_i(\mathbf{x}) = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x}) \tag{2}$$

We guess $v_i(\mathbf{x}) = a \in \mathbb{F}_q$. We have now:

$$\begin{aligned} p_i(\mathbf{x}) &= u_i(\mathbf{x}) + a \times w_i(\mathbf{x}) \\ v_i(\mathbf{x}) &= a \end{aligned}$$

$\hookrightarrow$ $m - 1$ polynomials in $n - 2$ variables.

Direct attack algorithm

- Guess $n/2$ linear equations
- Get the $n/2$ other
- Complexity : $n^3 q^{\frac{n}{2}}$

New Hybrid Approach

Hybrid method [Bettale et al., ACM, 2012]

- Guess an optimal k variables.
- Groebner basis algorithm on $m - k$ polynomials and $n - 2k$ variables.
- Asymptotic complexity at m/n and q fixed.

New Hybrid Approach

Hybrid method [Bettale et al., ACM, 2012]

- Guess an optimal k variables.
- Groebner basis algorithm on $m-k$ polynomials and $n-2k$ variables.
- Asymptotic complexity at m/n and q fixed.

Asymptotic complexity with $m = n$ and $q = 16$

- Classic : $2^{2.01n}$
- Biscuit polynomials : $2^{1.59n}$

New Hybrid Approach

Hybrid method [Bettale et al., ACM, 2012]

- Guess an optimal k variables.
- Groebner basis algorithm on $m - k$ polynomials and $n - 2k$ variables.
- Asymptotic complexity at m/n and q fixed.

Asymptotic complexity with $m = n$ and $q = 16$

- Classic : $2^{2.01n}$
- Biscuit polynomials : $2^{1.59n}$

Key recovery cost for Biscuit (`MQ-estimator`)

name	Claimed security level	Our attack
`biscuit128`	160	124
`biscuit192`	210	163
`biscuit256`	276	215

Forgery attack

Forgery attack

- Kales-Zaverucha forgery attack [Kales et al., Cham, 20].
- Solving a chosen polynomial subsystem.
  $\hookrightarrow$ easier in our case

Forgery attack

Forgery attack

- Kales-Zaverucha forgery attack [Kales et al., Cham, 20].
- Solving a chosen polynomial subsystem.
  $\hookrightarrow$ easier in our case

Security estimate

name	`biscuit128s`	`biscuit128f`
Claimed key-recovery cost	160	160
our attack	124	124
Claimed forgery cost	143	143
our attack	116	120

`biscuit128s` : $n = 64, m = 67, q = 16, N = 256,$
`biscuit128f` : $n = 64, m = 67, q = 16, N = 256$

Forgery attack

Interesting case

If the subsystem is underdetermined :

- $n - u$ polynomials in n variables
- We can add u linear dependencies

Forgery attack

Interesting case

If the subsystem is underdetermined :

- $n - u$ polynomials in n variables
- We can add u linear dependencies

Algorithm in this case

- With $i \in \{1, \ldots, u\}$, we set :
  $v_i(\mathbf{x}) = 0$
- $p_i = u_i(\mathbf{x}) + v_i(\mathbf{x}) \times w_i(\mathbf{x})$ becomes :
  $u_i(\mathbf{x}) = 0$

$\hookrightarrow$ We have now $n - 2u$ polynomials in $n - 2u$ variables to solve.

New parameters for Biscuit

Actual parameters

biscuit128s : $n = 64, m = 67, q = 16, N = 256$
$\rightarrow$ sig = 4 758 bytes

New parameters for Biscuit

Actual parameters

biscuit128s : $n = 64, m = 67, q = 16, N = 256$
$\rightarrow$ sig = 4 758 bytes

New parameters

q \ N	256	512	1024
16	$n = 80, m = 94$ sign = 5840	$n = 84, m = 104$ sign = 5730	$n = 80, m = 104$ sign = 5420
32	$n = 68, m = 77$ sign = 5910	$n = 70, m = 77$ sign = 5730	$n = 68, m = 77$ sign = 5470
256	$n = 47, m = 51$ sign = 6080	$n = 49, m = 55$ sign = 5890	$n = 47, m = 51$ sign = 5610

Work in progress

LWE with binary error

$A * s + e = b$ with

- $s \in \mathbb{F}_q^n$ the secret.
- $e \in \{0,1\}^m$ an unknown error vector.
- $A \in \mathbb{F}_q^{m \times n}$ and $b \in \mathbb{F}_q^m$ public.

Work in progress

LWE with binary error

$A * s + e = b$ with

- $s \in \mathbb{F}_q^n$ the secret.
- $e \in \{0, 1\}^m$ an unknown error vector.
- $A \in \mathbb{F}_q^{m \times n}$ and $b \in \mathbb{F}_q^m$ public.

Attack idea

- We have : $(\langle A_i, s \rangle - b_i)(\langle A_i, s \rangle - b_i - 1) = 0$
  $\hookrightarrow$ Quadratic polynomial in n variables over $\mathbb{F}_q$.
- We guess an optimal k e_i and solve $m - k$ polynomials of $n - k$ variables over $\mathbb{F}_q$.

Thank you !