THE MMAP STRIKES BACK:
Obfuscation and New Multilinear Maps
Immune to CLT13 Zeroizing Attacks
Fermi Ma and Mark Zhandry

RETURN OF GGH15:
Provable Security Against Zeroizing Attacks
James Bartusek, Jiaxin Guan, Fermi Ma, and Mark Zhandry
Multilinear Maps

Levels: $1, \ldots, \kappa$, Plaintext Ring $R$

**Secret**

\[ a \in R, \ i \in \{1, \ldots, \kappa\} \]

\[ \text{Encode} \rightarrow [a]_i \]

**Public**

\[ [a]_i + [b]_i \rightarrow [a + b]_i \]

\[ [a]_i \times [b]_j \rightarrow [ab]_{i+j} \]

\[ [a]_\kappa \rightarrow \text{Zero-Test} \rightarrow \text{Yes/No} \]
Indistinguishability Obfuscation (iO)

Bounded-Collusion Functional Encryption

Broadcast Encryption

Multilinear Maps
[GGH13,CLT13,GGH15]

Witness Encryption

Multiparty NIKE
Multilinear Maps

[GGH13, CLT13, GGH15]
CLT13 Maps

plaintext

\( m_i + r_i g_i \)

secret mask

Chinese Remainder Theorem

\( a \pmod{N} \)

“small” secret prime

“small” random
CLT13 Maps

plaintext $\left( \ldots, m_i + r_i g_i, \ldots \right)$

"small" secret prime

secret mask

"small" random

Chinese Remainder Theorem

"zero-test parameter"

Zero Test: $\rho_{ZT} \cdot a \pmod{N} \ll N$?
Zeroizing Attack on CLT13 [CHLRS15]

Setting

\[
b^{(1)} = \left( \ldots, \frac{B_i^{(1)}}{z}, \ldots \right), \quad b^{(2)} = \left( \ldots, \frac{B_i^{(2)}}{z}, \ldots \right)
\]

\[
a^{(1)}, \ldots, a^{(n)}, c^{(1)}, \ldots, c^{(n)}
\]

Where each \(a^{(i)} \cdot b^{(j)} \cdot c^{(k)}\) is encoding of zero
Zeroizing Attack on CLT13 [CHLRS15]

**Setting**

\[
b^{(1)} = \left( \ldots, \frac{B_i^{(1)}}{z}, \ldots \right), \quad b^{(2)} = \left( \ldots, \frac{B_i^{(2)}}{z}, \ldots \right)
\]

\[
a^{(1)}, \ldots, a^{(n)}, c^{(1)}, \ldots, c^{(n)}
\]

Where each \( a^{(i)} \cdot b^{(j)} \cdot c^{(k)} \) is encoding of zero

**Attack Steps**

1. Form matrices \( W, Y \) by zero-testing each \( a^{(i)} \cdot b^{(j)} \cdot c^{(k)} \).
2. Compute eigenvalues of \( W^{-1}Y \):

\[
\ldots, \frac{B_i^{(2)}}{z}, \ldots
\]

3. GCD on eigenvalues reveal secret parameters.
**Observation:** CHLRS15 computes char-poly(M) where entries of M are zero-test results. Roots are numerators $a_i + r_ig_i$.

**Attack Steps**

1. Form matrices $W, Y$ by zero-testing each $a^{(i)} \cdot b^{(j)} \cdot c^{(k)}$.
2. Compute eigenvalues of $W^{-1}Y$:
   
   $$
   B_i^{(1)}, ..., B_i^{(2)}, ...
   $$

3. GCD on eigenvalues reveal secret parameters.
Observation: CHLRS15 computes char-poly(M) where entries of M are zero-test results. Roots are numerators $a_i + r_i g_i$.

Solving polynomial for CLT13 numerators is only known attack strategy. [See also: CGHLMMRST15, CLLT16]

Attack Steps
1. Form matrices $W, Y$ by zero-testing each $a^{(i)} \cdot b^{(j)} \cdot c^{(k)}$.
2. Compute eigenvalues of $W^{-1}Y$:
   \[
   B_i^{(2)}, ..., \frac{B_i^{(2)}}{B_i^{(1)}}, ...
   \]
3. GCD on eigenvalues reveal secret parameters.
Observation: CHLRS15 computes $\text{char-poly}(M)$ where entries of $M$ are zero-test results. Roots are numerators $a_i + r_i g_i$.

Solving polynomial for CLT13 numerators is only known attack strategy. [See also: CGHLMMRST15, CLLT16]

\[
\begin{align*}
Q(\{t_j\}_j, \{s_i\}_i) &= 0 \\
Q(\{t_j\}_j, \{s_i\}_i) &\neq 0
\end{align*}
\]
Step 1: Weak Model

(INSPIRED BY MSZ16 AND GMMSSZ16)

Extend Generic Model to allow adversary to perform a zeroizing attack.
Step 1: Weak Model

(-inspired by MSZ16 and GMMSSZ16)

Extend Generic Model to allow adversary to perform a zeroizing attack.

Generic Model

Plaintexts $m^{(1)}, \ldots, m^{(k)}$.

Handles $h^{(1)}, \ldots, h^{(k)}$.

Zero Test Queries Return “zero” if

- $p(\{m^{(i)}\}_i) = 0$
- degree $\kappa$.
Extend Generic Model to allow adversary to perform a zeroizing attack.

Generic Model + Zeroizing Attacks

Plaintexts $m^{(1)}, \ldots, m^{(k)}$.
Handles $h^{(1)}, \ldots, h^{(k)}$.

Zero Test Queries Return “zero” if
- $p(\{m^{(i)}\}_i) = 0$
- degree $\kappa$.

New: Return post-zero-test handle “$T$” if zero.

Post Zero Test Return “WIN” if
- $Q(\{t_j\}_j, \{s_i\}_i) = 0$
- $Q(\{t_j\}_j, \{S_i\}_i) \neq 0$
Step 1: Weak Model

Extend Generic Model to allow adversary to perform a zeroizing attack.

Step 2: Annihilation Theorem

If you can perform a zeroizing attack, you can annihilate “zero-test polynomials”.
Step 1: Weak Model

Extend **Generic Model** to allow adversary to perform a zeroizing attack.

Step 2: Annihilation Theorem

If you can perform a zeroizing attack, you can annihilate “zero-test polynomials”.

If $x, y$ are CLT13 encodings, and $x^2 + xy$ is a top-level zero, **the zero-test polynomial** is the formal polynomial $x^2 + xy$.

**Theorem**: If can mount a zeroizing attack, can “cancel out” linearly independent zero-test polynomials.
<table>
<thead>
<tr>
<th>Step 1: Weak Model</th>
<th>Extend Generic Model to <em>allow adversary</em> to perform a zeroizing attack.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2: Annihilation Theorem</td>
<td>If you can perform a zeroizing attack, you can annihilate “zero-test polynomials”.</td>
</tr>
<tr>
<td>Step 3: Zeroizing-Immune Schemes</td>
<td>Obtain constructions where annihilating zero-test polynomials is hard.</td>
</tr>
</tbody>
</table>
Extend Generic Model to allow adversary to perform a zeroizing attack.

If you can perform a zeroizing attack, you can annihilate “zero-test polynomials”.

Obtain constructions where annihilating zero-test polynomials is hard.

- For BMSZ16 Obfuscation and BLRSZZ16 ORE it is provably hard to annihilate zero-test polynomials (from standard assumptions [GMMSSZ16])
- New multilinear map hard to annihilate (under new non-standard assumption).
GGH15
Construction
GGH15
Construction
GGH15 Construction
\[ A_u \xrightarrow{Enc} D \xrightarrow{} A_v \xrightarrow{} A_w \]

\[ A_u \cdot D = S \cdot A_v + E \]

**GGH15 Construction**
GGH15 Construction
GGH15 Construction
$$A_u \xrightarrow{S_1} D_1 \xrightarrow{Enc} A_v \xrightarrow{S_2} D_2 \xrightarrow{Enc} A_w$$

GGH15 Construction
GGH15 Construction
\[ A_u \quad S_1 \xrightarrow{\text{Enc}} \quad D_1 \quad A_v \quad S_2 \xrightarrow{\text{Enc}} \quad D_2 \quad A_w \]

\[ A_u \begin{bmatrix} D_1 & D_2 \end{bmatrix} = \begin{bmatrix} S_1 & S_2 & A_w \end{bmatrix} + \begin{bmatrix} E_2 \end{bmatrix} + \begin{bmatrix} E_1 \end{bmatrix} \]

**GGH15 Construction**
 GGH15 Construction
GGH15 Construction
$S_1 \ S_2 = 0$

GGH15 Construction
Toy Attack
[CLLT16, CGH17, CVW18]
$Z_{T_{i,j}} := \text{top left entry of } A_u D_i D_j$
\[ A_u D_i D_j = S_i S_x A_w + S_i E_j + E_i D_j \]

short

\[ ZT_{1,2} ZT_{1,4} = \]

\[ ZT_{3,2} ZT_{3,4} = \]

\[ S_1 S_2 = 0 \]
\[ S_3 S_4 = 0 \]
Vector in left kernel gives algebraic relation on secrets!
Multilinear Maps

This Talk: Algebraic Zeroizing Attacks
Multilinear Maps

[GGH13, CLT13, GGH15]

[HuJia15]

[MSZ16]

[CHLRS15]

[CLLT16]

[CGH17]

[CVW18]

[Hal15]

[CGHLMRST]

This Talk: Algebraic Zeroizing Attacks

Algebraic Zeroizing Attack

Statistical Zeroizing Attack?
Statistical Zeroizing Attack: Cryptanalysis of Candidates of BP Obfuscation over GGH15 Multilinear Map

Jung Hee Cheon, Wonhee Cho, Minki Hhan, Jiseung Kim, and Changmin Lee

ePrint: 2018/1081

First polynomial-time, non-algebraic zeroizing attack on GGH15-based obfuscation!
Note: We temporarily add the disclaimer not to mislead the readers and audiences of TCC.

Disclaimer

The authors of BGMZ obfuscation [4] (TCC’18) report that there are flaws of cryptanalysis of BGMZ obfuscation in Section 5. In particular, the current optimal parameter choice of BGMZ obfuscation is robust against our attack, while the attack lies outside the provable security of BGMZ obfuscation.

The flaws in the analysis in Section 5 are as follows:

- $\nu$ is chosen to $\text{poly}(\lambda)$ in this paper whereas the original paper [4] chooses $\nu = 2^\lambda$ (or at least super-polynomial of $\lambda$).

- The analysis of our attack claims that $(1 + \frac{2}{g})^h$ is polynomial of $\lambda$, but it is not true since $g = 5$ is constant.

We remark that our attack gives a constraint on the parameters; BGMZ obfuscation with $\sigma = \exp(\lambda)^a$ can be broken in the same manner with slightly modified proof. We will update the paper as soon as possible.

$^a$ Interestingly, this choice gives a countermeasure of CVW obfuscation.
Extend \textbf{Generic Model} to \textit{allow adversary} to perform an algebraic zeroizing attack.

\textbf{GGH15 Algebraic Zeroizing Model}

- \textbf{Graph} $G$, \textbf{Plaintexts} $\{S_i, u_i \rightarrow v_i\}$
- $D_i \leftarrow \text{Enc}(S_i, u_i \rightarrow v_i)$
- \textbf{Handles} $\{h_i \rightarrow (D_i, S_i, u_i \rightarrow v_i)\}$

\textbf{Zero Test Queries}: if
- $p_j$ edge-respecting
- $p_j(\{D_i\}_i) = (T_j, \text{“is zero”})$

Return \textit{post-zero-test} handle \textbf{“$T_j$”}

\textbf{Post Zero Test} Return \textit{“WIN”} if
- $Q(\{T_j\}, \{S_i\}) = 0$
- $Q(\{T_j\}, \{S_i\}) \neq 0$, $Q(\{T_j\}, \{S_i\}) \neq 0$
Our GGH15 Variant

\[ D_{u \rightarrow v} \]
Our GGH15 Variant

$R_u^{-1}$  $D_{u \rightarrow v}$  $R_v$

$B$
Our GGH15 Variant

B injects entropy

algebraic relation involving \( \{T_j\} \) \rightarrow \text{annihilation of zero-test polynomials } \{p_j\}
Our GGH15 Variant

GGH15 Annihilation Theorem

hardness of annihilating zero-test polynomials → security in our model!
Branching Program (BP) Obfuscation

$f \rightarrow S_{1,1} \ S_{2,1} \ S_{3,1} \ S_{4,1} \ S_{5,1} \ S_{6,1} \ S_{1,2} \ S_{2,2} \ S_{3,2} \ S_{4,2} \ S_{5,2} \ S_{6,2}$

$f(x) = 1 \leftrightarrow \prod_{i} S_{i,x_{inp}(i)} = 0$

Simple Obfuscation Construction:
Encode $S_{i,b}$ matrices with our new GGH15 variant
Zeroizing Attack on BP Obfuscation

Simple Obfuscation Construction:
Encode $S_{i,b}$ matrices with our new GGH15 variant
Zeroizing Attack on BP Obfuscation

GGH15 Annihilation Theorem

Annihilation of Successful Zero-Test Polynomials

Simple Obfuscation Construction:
Encode $S_{i,b}$ matrices with our new GGH15 variant
Zeroizing Attack on BP Obfuscation

GGH15 Annihilation Theorem

Annihilation of Successful Zero-Test Polynomials

\( p \)-Bounded Speedup Hypothesis [MSW14]

Annihilation of (read many) BP evaluations

Simple Obfuscation Construction:
Encode \( S_{i,b} \) matrices with our new GGH15 variant
Zeroizing Attack on BP Obfuscation

GGH15 Annihilation Theorem

Annihilation of Successful Zero-Test Polynomials

p-Bounded Speedup Hypothesis [MSW14]

Annihilation of (read many) BP evaluations

Distinguisher for any PRF in NC1

Simple Obfuscation Construction:
Encode $S_{i,b}$ matrices with our new GGH15 variant
Zeroizing Attack on BP Obfuscation

GGH15 Annihilation Theorem

Annihilation of Successful Zero-Test Polynomials

$p$-Bounded Speedup Hypothesis [MSW14]

Annihilation of (read many) BP evaluations

Distinguisher for any PRF in NC1

Simple Obfuscation Construction:
Encode $S_{i,b}$ matrices with our new GGH15 variant

Thank you! Questions?