Bitslice Masking and Improved Shuffling: How and When to Mix Them in Software?

Melissa Azouaoui, Olivier Bronchain, Vincent Grosso
Kostas Papagiannopoulos, François-Xavier Standaert

CHES 2022, Leuven, Belgium
Contents

Introduction

Linear layers

Non-linear layers

Perf. vs security
Countermeasures compared on:

- Run time overheads.
- Worst-case security ($N$).
Design space for side-channel countermeasures

Countermeasures compared on:
- Run time overheads.
- Worst-case security ($N$).

Worst-case security ($N$) vs Overheads
Design space for side-channel countermeasures

Countermeasures compared on:
- Run time overheads.
- Worst-case security ($N$).

Design 1
Design 2
Design space for side-channel countermeasures

Countermeasures compared on:
- Run time overheads.
- Worst-case security ($N$).

$\rightarrow$ Best design depends on desired security.
Design space for side-channel countermeasures

Countermeasures compared on:

- Run time overheads.
- Worst-case security ($N$).

→ Best design depends on desired security.

Best design is device dependent:

- Noise level.
- Platform architecture.
Design space for side-channel countermeasures

Countermeasures compared on:
- Run time overheads.
- Worst-case security ($N$).

→ Best design depends on desired security.

Best design is device dependent:
- Noise level.
- Platform architecture.
Existing side-channel countermeasures \((\text{MI}(X; L) < 1)\)

**Masking:**

- Randomized the data processed.
- Sharing of \(x := (x^0, x^1, \ldots, x^{d-1})\)
- Noise amplification:

\[
N \approx \frac{c}{\prod_{i} \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
\]

- Data Layout:

<table>
<thead>
<tr>
<th>0-th share</th>
<th>1-th share</th>
<th>2-th share</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x^0)</td>
<td>(x^1)</td>
<td>(x^2)</td>
</tr>
</tbody>
</table>
**Existing side-channel countermeasures**

\[
\text{Mascking:} \quad (\text{MI}(X; L) < 1)
\]

- Randomized the data processed.
- Sharing of \( x := (x^0, x^1, \ldots, x^{d-1}) \)
- Noise amplification:

\[
N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
\]

- Data Layout:

  0-th share
  
  1-th share
  
  2-th share

**Shuffling:**

- Randomized processing order.
- Execution based on a perm. of size \( \eta \)
- Noise addition:

\[
N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
\]

- Data Layout:

  Perm:

  data:

  indep. data
Existing side-channel countermeasures \( (\text{MI}(X; L) < 1) \)

Masking:
- Randomized the data processed.
- Sharing of \( x := (x^0, x^1, \ldots, x^{d-1}) \)
- Noise amplification:
  \[
  N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
  \]
- Data Layout:
  
<table>
<thead>
<tr>
<th>0-th share</th>
<th>1-th share</th>
<th>2-th share</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x^0 )</td>
<td>( x^1 )</td>
<td>( x^2 )</td>
</tr>
</tbody>
</table>

Shuffling:
- Randomized processing order.
- Execution based on a perm. of size \( \eta \)
- Noise addition:
  \[
  N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
  \]
- Data Layout:
  
  Perm:
  
  | 2 | 4 | 1 | 3 | 0 |

  data:
  
  \[
  \begin{array}{c|c|c}
  \hline
  \text{indep. data} & \hline
  \\hline
  \end{array}
  \]
# Existing side-channel countermeasures

$$\text{Performance vs Security} \quad \left(\text{MI}(X; L) < 1\right)$$

## Masking:
- Randomized the data processed.
- Sharing of $$x := (x^0, x^1, \ldots, x^{d-1})$$
- Noise amplification:
  \[
  N \approx \frac{c}{\Pi_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
  \]
- Data Layout:

<table>
<thead>
<tr>
<th>Share</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-th share</td>
<td>$$x^0$$</td>
</tr>
<tr>
<td>1-th share</td>
<td>$$x^1$$</td>
</tr>
<tr>
<td>2-th share</td>
<td>$$x^2$$</td>
</tr>
</tbody>
</table>

## Shuffling:
- Randomized processing order.
- Execution based on a perm. of size $$\eta$$
- Noise addition:
  \[
  N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
  \]
- Data Layout:

<table>
<thead>
<tr>
<th>Perm:</th>
<th>Data:</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 4 1 3 0</td>
<td>$$x_2$$ $$x_4$$</td>
</tr>
<tr>
<td>indep. data</td>
<td></td>
</tr>
</tbody>
</table>
**Existing side-channel countermeasures** \((\text{MI}(X; L) < 1)\)

**Masking:**
- Randomized the data processed.
- Sharing of \(x := (x^0, x^1, \ldots, x^{d-1})\)
- Noise amplification:

\[
N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
\]

- Data Layout:
  - 0-th share: \(x^0\)
  - 1-th share: \(x^1\)
  - 2-th share: \(x^2\)

**Shuffling:**
- Randomized processing order.
- Execution based on a perm. of size \(\eta\)
- Noise addition:

\[
N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
\]

- Data Layout:
  - Perm: \(2 \ 4 \ 1 \ 3 \ 0\)
  - data: \(x_1 \ x_2 \ x_4\)
  - indep. data

\(\text{MI}(X; L) < 1\)
Existing side-channel countermeasures

\((\text{MI}(X; L) < 1)\)

Masking:
- Randomized the data processed.
- Sharing of \(x := (x^0, x^1, \ldots, x^{d-1})\)
- Noise amplification:
  \[ N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d} \]
- Data Layout:
  
<table>
<thead>
<tr>
<th>0-th share</th>
<th>1-th share</th>
<th>2-th share</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x^0)</td>
<td>(x^1)</td>
<td>(x^2)</td>
</tr>
</tbody>
</table>

Shuffling:
- Randomized processing order.
- Execution based on a perm. of size \(\eta\)
- Noise addition:
  \[ N \approx \frac{\eta \cdot c}{\text{MI}(X; L)} \]
- Data Layout:
  
  Perm:
  
  | 2 | 4 | 1 | 3 | 0 |
  
  data:
  
  \(x_1 \ x_2 \ x_3 \ x_4\)

\(\text{indep. data}\)
Existing side-channel countermeasures \((\text{MI}(X; L) < 1)\)

**Masking:**
- Randomized the data processed.
- Sharing of \(x := (x^0, x^1, \ldots, x^{d-1})\)
- Noise amplification:
  \[
  N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{c}{\text{MI}(X^i; L)^d}
  \]
- Data Layout:
  
<table>
<thead>
<tr>
<th>Share</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-th</td>
<td>(x^0)</td>
</tr>
<tr>
<td>1-th</td>
<td>(x^1)</td>
</tr>
<tr>
<td>2-th</td>
<td>(x^2)</td>
</tr>
</tbody>
</table>

**Shuffling:**
- Randomized processing order.
- Execution based on a perm. of size \(\eta\)
- Noise addition:
  \[
  N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
  \]
- Data Layout:
  
  Perm:
  
  \[
  2 \quad 4 \quad 1 \quad 3 \quad 0
  \]

  data:
  
  \[
  x^0 \quad x^1 \quad x^2 \quad x^3 \quad x^4
  \]

  indep. data
Existing side-channel countermeasures \((\text{MI}(X; L) < 1)\)

**Masking:**
- Randomized the data processed.
- Sharing of \(x := (x^0, x^1, \ldots, x^{d-1})\)
- Noise amplification:

\[
N \approx \frac{c}{\prod_i \text{MI}(X^i; L)} \approx \frac{d}{\text{MI}(X^i; L)^d}
\]

- Data Layout:
  - 0-th share: \(x^0\)
  - 1-th share: \(x^1\)
  - 2-th share: \(x^2\)

**Shuffling:**
- Randomized processing order.
- Execution based on a perm. of size \(\eta\)
- Noise addition:

\[
N \approx \frac{\eta \cdot c}{\text{MI}(X; L)}
\]

- Data Layout:
  - Perm: \(\text{indep. data}\)
  - data: \(x_0, x_1, x_2, x_3, x_4\)

→ How to amplify shuffling thanks to masking? \((\eta^d)\)
Design space for side-channel countermeasures

1. Security:
   ▶ Explore design space for shuffling + masking.
   ▶ Evaluate the security:
     ▶ Paper & pencil.
     ▶ Confirmed with simulations.

Rivain et al. [RPD09]:
   ▶ Linear layers: \( \binom{d \cdot \eta}{d} \)
   ▶ Non-linear layers: \( \eta \)
Design space for side-channel countermeasures

1. Security:
   - Explore design space for shuffling + masking.
   - Evaluate the security:
     - Paper & pencil.
     - Confirmed with simulations.

Rivain et al. [RPD09]:
   - Linear layers: $d \cdot \eta$
   - Non-linear layers: $\eta$

2. Performances:
   - Explore perf. bitslice and shuffle.
   - Benchmarks on Cortex-M4.
Design space for side-channel countermeasures

1. Security:
   ▶ Explore design space for shuffling + masking.
   ▶ Evaluate the security:
     ▶ Paper & pencil.
     ▶ Confirmed with simulations.

Rivain et al. [RPD09]:
   ▶ Linear layers: $\binom{d \cdot \eta}{d}$
   ▶ Non-linear layers: $\eta$

2. Performances:
   ▶ Explore perf. bitslice and shuffle.
   ▶ Benchmarks on Cortex-M4.

3. Performances vs. security:
   ▶ Pertinence of masking and shuffling combination.
Contents

Introduction

Linear layers

Non-linear layers

Perf. vs security
Protecting masked linear layers

\[
\begin{align*}
\mathbf{x}_0 & \quad \mathbf{x}_1 & \quad \mathbf{x}_2 & \quad \mathbf{x}_3 & \quad \mathbf{x}_4 \\
\mathbf{x}_0^1 & \quad \mathbf{x}_1^1 & \quad \mathbf{x}_2^1 & \quad \mathbf{x}_3^1 & \quad \mathbf{x}_4^1 \\
\mathbf{x}_0^2 & \quad \mathbf{x}_1^2 & \quad \mathbf{x}_2^2 & \quad \mathbf{x}_3^2 & \quad \mathbf{x}_4^2 \\
\end{align*}
\]

\[
\begin{align*}
\mathbf{y}_0 & \quad \mathbf{y}_1 & \quad \mathbf{y}_2 & \quad \mathbf{y}_3 & \quad \mathbf{y}_4 \\
\mathbf{y}_0^1 & \quad \mathbf{y}_1^1 & \quad \mathbf{y}_2^1 & \quad \mathbf{y}_3^1 & \quad \mathbf{y}_4^1 \\
\mathbf{y}_0^2 & \quad \mathbf{y}_1^2 & \quad \mathbf{y}_2^2 & \quad \mathbf{y}_3^2 & \quad \mathbf{y}_4^2 \\
\end{align*}
\]

Setting:

- \( \eta = 5 \) independent data \( x_j \) and \( y_j \)
- \( d = 3 \) shares \( x^i \) and \( y^i \).
Protecting masked linear layers

Setting:
- $\eta = 5$ independent data $x_j$ and $y_j$
- $d = 3$ shares $x^i$ and $y^i$.

Goal:
- Compute all: $z^i_j = x^i_j \oplus y^i_j$
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

$z_1^3 = x_1^3 \oplus y_1^3$

$x_0$  $x_1$  $x_2$

$y_0$  $y_1$  $y_2$

$z_0$  $z_1$  $z_2$

indep. data

indep. shares
**Shuffling-tuples on linear layers: description**

**Description:**
- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

$$z_2^3 = x_2^3 \oplus y_2^3$$
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

$z_0^4 = x_0^4 \oplus y_0^4$

Decomposition: Decrease $MI(X; L)$ by a factor $\eta$. $N \approx c \cdot \eta Q_i MI(X_i; L) \rightarrow$ Masking does not amplify shuffling.
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

\[ z_1^4 = x_1^4 \oplus y_1^4 \]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

$z_2^4 = x_2^4 \oplus y_2^4$
Shuffling-tuples on linear layers: description

Description:
- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: \( \eta \).
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: \( \eta \).

\[
\begin{align*}
L_{z}^{3} &= x_{0}^{3} \oplus y_{0}^{3} \\
L_{z}^{2} &= x_{1}^{2} \oplus y_{1}^{2} \\
L_{z}^{1} &= x_{2}^{1} \oplus y_{2}^{1} \\
L_{z}^{0} &= x_{0}^{0} \oplus y_{0}^{0}
\end{align*}
\]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

$$L_{z_0} = x_0^0 \oplus y_0^0$$
$$L_{z_1} = x_1^1 \oplus y_1^1$$
$$L_{z_2} = x_2^2 \oplus y_2^2$$
$$L_{z_3} = x_3^3 \oplus y_3^3$$
$$L_{z_4} = x_4^4 \oplus y_4^4$$
Shuffling-tuples on linear layers: description

Description:
- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

\[
x_0^0 \quad x_0^1 \quad x_0^2 \quad x_0^3 \quad x_0^4 \\
x_1^0 \quad x_1^1 \quad x_1^2 \quad x_1^3 \quad x_1^4 \\
x_2^0 \quad x_2^1 \quad x_2^2 \quad x_2^3 \quad x_2^4 \\
\]
\[
z_0^1 = x_0^1 \oplus y_0^1 \\
\]
\[
y_0^0 \quad y_0^1 \quad y_0^2 \quad y_0^3 \quad y_0^4 \\
y_1^0 \quad y_1^1 \quad y_1^2 \quad y_1^3 \quad y_1^4 \\
y_2^0 \quad y_2^1 \quad y_2^2 \quad y_2^3 \quad y_2^4 \\
\]
\[
z_1^0 \quad z_1^1 \quad z_1^2 \quad z_1^3 \quad z_1^4 \\
z_2^0 \quad z_2^1 \quad z_2^2 \quad z_2^3 \quad z_2^4 \\
\]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

\[ z^1_0 = x^1_0 \oplus y^1_0 \]

\[ z^1_1 = x^1_1 \oplus y^1_1 \]

\[ z^1_2 = x^1_2 \oplus y^1_2 \]

\[ z^1_3 = x^1_3 \oplus y^1_3 \]

\[ z^1_4 = x^1_4 \oplus y^1_4 \]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

\[
\begin{align*}
L_3 &= x_3^0 + y_3^0 \\
L_4 &= x_4^0 + y_4^0 \\
L_1 &= x_1^1 + y_1^1 \\
L_2 &= x_2^1 + y_2^1
\end{align*}
\]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

$$z_0^0 = x_0^0 \oplus y_0^0$$

$$z_0^1 = x_0^1 \oplus y_0^1$$

$$z_0^2 = x_0^2 \oplus y_0^2$$
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$. 

\[ z_1^0 = x_1^0 \oplus y_1^0 \]
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

Decrease $\text{MI}(X; L)$ by a factor $\eta$. 

$$N \approx \frac{c \cdot \eta}{\prod_i \text{MI}(X^i; L)}$$

→ Masking does not amplify shuffling.
Shuffling-tuples on linear layers: description

Description:

- Shuffle between variables
- Permutations:
  - Number: 1.
  - Size: $\eta$.

Decrease $\text{MI}(X; L)$ by a factor $\eta$.

$$N \approx \frac{c \cdot \eta}{\prod_i \text{MI}(X_i; L)}$$

→ Masking does not amplify shuffling.
Shuffling-shares on linear layers: simulations

Expected security:
\[ N \approx \frac{c \cdot \eta}{\prod_i \text{MI}(X^i; L)} \]
Shuffling-shares on linear layers: description

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

Description:

$z_0^1 = x_0^1 \oplus y_0^1$

$z_0 \oplus z_1$
Shuffling-shares on linear layers: description

Description:

- Shuffle the \( i \)-th share of each \( x_j \).
- Permutations:
  - Number: \( d \).
  - Size: \( \eta \).

\[
\begin{align*}
&x_0^0, x_1^0 \\
\hline
&\text{indep. data} \\
&\text{indep. shares} \\
&y_0^0, y_1^0 \\
\hline
&z_0^0 = x_0^0 \oplus y_0^0
\end{align*}
\]

Masking amplifies shuffling.

\[
\text{MI} = \frac{\text{MI}_i(X_i; L(X_\eta))}{\eta} \approx \text{MI} \cdot (X_\eta; iL_d)
\]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[
\begin{align*}
  z_0^2 &= x_0^2 \oplus y_0^2 \\
  z_0 &= x_0 \oplus z_0^2
\end{align*}
\]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[ z_0^3 = x_0^3 \oplus y_0^3 \]
Shuffling-shares on linear layers: description

Description:
- Shuffle the \( i \)-th share of each \( x_j \).
- Permutations:
  - Number: \( d \).
  - Size: \( \eta \).

\[
\begin{align*}
1 & \quad 0 & \quad 2 & \quad 3 & \quad 4 \\
x_0^0 & x_1^0 & x_2^0 & x_3^0 & x_4^0 \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
\end{align*}
\]

\[
\begin{align*}
z_0^4 = x_0^4 \oplus y_0^4 \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
\end{align*}
\]

\[
\begin{align*}
y_0^0 & y_1^0 & y_2^0 & y_3^0 & y_4^0 \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
& & & & \\
\end{align*}
\]
Shuffling-shares on linear layers: description

<table>
<thead>
<tr>
<th>4</th>
<th>0</th>
<th>1</th>
<th>3</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_0 )</td>
<td>( x_1 )</td>
<td>( x_2 )</td>
<td>( x_3 )</td>
<td>( x_4 )</td>
</tr>
<tr>
<td>( x_4 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ z_1^4 = x_1^4 \oplus y_1^4 \]

**Description:**

- Shuffle the \( i \)-th share of each \( x_j \).
- Permutations:
  - Number: \( d \).
  - Size: \( \eta \).

\[ z_0^0 \quad z_0^1 \quad z_1^0 \quad z_2^0 \quad z_3^0 \quad z_4^0 \]

\[ z_1^1 \]

\[ y_0^0 \quad y_1^0 \quad y_2^0 \quad y_3^0 \quad y_4^0 \]

\[ y_4^1 \]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[ z_1^0 = y_1^0 + y_1^0 \]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[
\begin{array}{c}
4 & 0 & 1 & 3 & 2 \\
\hline
x_0^0 & x_1^0 & x_2^0 & x_3^0 & x_4^0 \\
x_0^1 & x_1^1 & & & \\
\hline
\end{array}
\]

\[
\begin{array}{c}
y_0^0 & y_1^0 & y_2^0 & y_3^0 & y_4^0 \\
y_0^1 & y_1^1 & & & \\
\hline
\end{array}
\]

\[
\begin{array}{c}
z_0^0 & z_1^0 & z_2^0 & z_3^0 & z_4^0 \\
z_0^1 & z_1^1 & & & \\
\hline
\end{array}
\]

\[
z_1^1 = x_1^1 \oplus y_1^1
\]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[
\begin{align*}
\tilde{z}_1^3 &= x_1^3 \oplus y_1^3 \\
y_0^0 & x_0^0 & y_0^0 & x_0^0 & x_4^0 \\
y_0^1 & y_1^1 & x_0^1 & x_3^1 & x_4^1 \\
y_1^1 & y_1^1 & y_3^1 & y_4^1 & y_4^1 \\
\end{align*}
\]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[ z_1^2 = x_1^2 \oplus y_1^2 \]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[ z_2^0 = x_2^0 \oplus y_2^0 \]

![Diagram showing shuffling-shares on linear layers](image)
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

$$z_2^2 = x_2^2 \oplus y_2^2$$
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

\[
\begin{align*}
z_2^1 &= x_2^1 \oplus y_2^1 \\

\begin{array}{c}
\begin{array}{cccccc}
0 & 2 & 1 & 4 & 3 \\
x_0^0 & x_1^0 & x_2^0 & x_3^0 & x_4^0 \\
x_0^1 & x_1^1 & x_2^1 & x_3^1 & x_4^1 \\
x_0^2 & x_1^2 & x_2^2 & x_3^2 & x_4^2 \\
\end{array}
\end{array}
\end{align*}
\]
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.
Shuffling-shares on linear layers: description

Description:

- Shuffle the $i$-th share of each $x_j$.
- Permutations:
  - Number: $d$.
  - Size: $\eta$.

Decrease $\text{MI}(X^i; L)$ by a factor $\eta$.

$$N \approx \frac{c}{\prod_i \text{MI}(X^i; L) / \eta} \approx \frac{c \cdot \eta^d}{\text{MI}(X^i; L)^d}$$

$\rightarrow$ Masking amplifies shuffling.
Shuffling-shares on linear layers: simulations

Expected security:

$$N \approx \frac{c \cdot \eta^d}{\text{MI}(X^i; L)^d}$$
Shuffling-everything on linear layers: description

Description:
- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$. 

\[ z_2^0 = x_2^0 \oplus y_2^0 \]

\[ z_0^2 \]
Shuffling-everything on linear layers: description

Description:
- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.
Shuffling-everything on linear layers: description

Description:
- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: \( d \cdot \eta \).
Shuffling-everything on linear layers: description

Description:
- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$. 

\[
\begin{align*}
\{0,1,4,9,10\} \\
\{5,2,14,3,8\} \\
\{12,6,7,11,13\}
\end{align*}
\]

\[
\begin{array}{c|c|c}
0 & 1 & 4 \\
2 & 9 & 10 \\
5 & 2 & 14 \\
3 & 8 & \hline \\
12 & 6 & 7 \\
11 & 13 & \hline
\end{array}
\]

\[
\begin{align*}
x^0_0 & \quad x^0_1 \quad x^0_4 \\
& \quad x^1_4 \\
x^2_0 & \\
\end{align*}
\]

\[
\begin{align*}
z^0_0 = x^0_0 \oplus y^0_0 \\
\end{align*}
\]

\[
\begin{array}{c|c|c}
0 & 1 & 4 \\
2 & 9 & 10 \\
5 & 2 & 14 \\
3 & 8 & \hline \\
12 & 6 & 7 \\
11 & 13 & \hline
\end{array}
\]

\[
\begin{array}{c|c|c}
y^0_0 & \quad y^0_1 \quad y^0_4 \\
& \quad y^1_4 \\
y^2_0 & \\
\end{array}
\]

\[
\begin{align*}
z^0_0 & \quad z^0_1 \quad z^0_4 \\
& \quad z^1_4 \\
z^2_0 & \\
\end{align*}
\]

\[
\begin{array}{c|c|c}
0 & 1 & 4 \\
2 & 9 & 10 \\
5 & 2 & 14 \\
3 & 8 & \hline \\
12 & 6 & 7 \\
11 & 13 & \hline
\end{array}
\]

\[
\begin{array}{c|c|c}
y^0_0 & \quad y^0_1 \quad y^0_4 \\
& \quad y^1_4 \\
y^2_0 & \\
\end{array}
\]

\[
\begin{align*}
z^0_0 & \quad z^0_1 \quad z^0_4 \\
& \quad z^1_4 \\
z^2_0 & \\
\end{align*}
\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.
Shuffling-everything on linear layers: description

**Description:**

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: \( d \cdot \eta \).

\[
\begin{align*}
\mathbf{z}_0^2 &= \mathbf{x}_0^2 \oplus \mathbf{y}_0^2 \\
\mathbf{z}_0 &= \mathbf{x}_0 \oplus \mathbf{y}_0
\end{align*}
\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.

\[
L_{z_0}^{0} = x_{0}^{0} \oplus y_{0}^{0}
\]

\[
L_{z_1}^{0} = x_{1}^{0} \oplus y_{0}^{0}
\]

\[
L_{z_2}^{0} = x_{2}^{0} \oplus y_{2}^{0}
\]

\[
L_{z_3}^{0} = x_{3}^{0} \oplus y_{3}^{0}
\]

\[
L_{z_4}^{0} = x_{4}^{0} \oplus y_{4}^{0}
\]

\[
z_{2}^{0} = x_{2}^{0} \oplus y_{2}^{0}
\]

\[
z_{2}^{1} = x_{2}^{1} \oplus y_{2}^{1}
\]

\[
z_{2}^{2} = x_{2}^{2} \oplus y_{2}^{2}
\]

\[
z_{2}^{3} = x_{2}^{3} \oplus y_{2}^{3}
\]

\[
z_{2}^{4} = x_{2}^{4} \oplus y_{2}^{4}
\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$. 

\[ z_0^3 = x_0^3 \oplus y_0^3 \]
Shuffling-everything on linear layers: description

- **Description:**
  - Shuffle all the possible operations.
  - **Permutations:**
    - Number: 1.
    - Size: $d \cdot \eta$.

---

\[ z_1^3 = x_1^3 \oplus y_1^3 \]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.

\[
z_2^2 = x_2^2 \oplus y_2^2\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$. 

\[
\begin{array}{cccccc}
10 & 1 & 4 & 9 & 0 \\
5 & 2 & 14 & 3 & 8 \\
12 & 6 & 7 & 11 & 13 \\
\end{array}
\]

\[
\begin{array}{cccc}
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\end{array}
\]

\[
x_0^0 \ x_0^1 \ x_0^2 \ x_1^0 \ x_1^1 \\
x_1^0 \ x_1^1 \ x_1^2 \ x_2^0 \\
x_2^0 \ x_2^1 \ x_2^2 \ x_3^0 \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\end{array}
\]

\[
\begin{array}{cccc}
z_0^0 & z_0^1 & z_0^2 & z_0^3 \\
z_1^0 & z_1^1 & z_1^2 & z_1^3 \\
z_2^0 & z_2^1 & z_2^2 & z_2^3 \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots \\
\end{array}
\]

\[
z_1^1 = x_1^1 \oplus y_1^1
\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$. 

\[
x_0^0, x_1^0, x_2^0, x_3^0, x_4^0 \\
x_0^1, x_1^1, x_2^1, x_3^1, x_4^1 \\
x_0^2, x_1^2, x_2^2, x_4^2
\]

\[
z_0^0, z_0^1, z_0^2, z_0^3, z_0^4 \\
z_1^0, z_1^1, z_1^2, z_1^3, z_1^4 \\
z_2^0, z_2^1, z_2^2, z_2^3 \\
z_4^0, z_4^1, z_4^2, z_4^3
\]

\[
z_1^2 = x_1^2 \oplus y_1^2
\]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: $d \cdot \eta$.

\[ z^1_2 = x^1_2 \oplus y^1_2 \]

\[ z^0_0, z^0_1, z^0_2, z^0_3, z^0_4 \]
\[ z^1_0, z^1_1, z^1_2, z^1_3, z^1_4 \]
\[ z^2_0, z^2_1, z^2_2, z^2_3, z^2_4 \]
Shuffling-everything on linear layers: description

Description:

- Shuffle all the possible operations.
- Permutations:
  - Number: 1.
  - Size: \( d \cdot \eta \).

\[ z_2^3 = x_2^3 \oplus y_2^3 \]

\[ N \approx c \cdot \binom{d \cdot \eta}{d} / \prod_i \text{MI}(X^i; L) \]

→ Masking amplifies shuffling.
Shuffling-everything on linear layers: simulations

$d = 2$

$d = 3$

Expected security:

$$N \approx \frac{c \cdot (d \cdot \eta)}{\prod_i \text{MI}(X^i; L)}$$
Contents

Introduction

Linear layers

Non-linear layers

Perf. vs security
Non-linear layers: summary of the results

For shuffled multiplications:

- Shuffling-shares and shuffling-tuples still apply with similar gain.
- Shuffling-everything could not be analyzed with paper & pencil:
  - Permutation on the output shares is not uniform.

<table>
<thead>
<tr>
<th></th>
<th>Linear layer</th>
<th></th>
<th></th>
<th>Non-linear layer</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gain</td>
<td></td>
<td></td>
<td>perm.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>shuffling-tuples</td>
<td>$\eta$</td>
<td>$\eta$</td>
<td>1</td>
<td>$\eta$</td>
<td>$\eta$</td>
<td>1</td>
</tr>
<tr>
<td>shuffling-shares</td>
<td>$\eta^d$</td>
<td>$\eta$</td>
<td>$d$</td>
<td>$\eta^d$</td>
<td>$\eta$</td>
<td>$d^2$</td>
</tr>
<tr>
<td>shuffling-everything</td>
<td>$(d\cdot\eta)_d$</td>
<td>$d \cdot \eta$</td>
<td>1</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

Table: Summary of the shuffling + masking combinations.
Non-linear layers: summary of the results

For shuffled multiplications:

- Shuffling-shares and shuffling-tuples still apply with similar gain.
- Shuffling-everything could not be analyzed with paper & pencil:
  - Permutation on the output shares is not uniform.

<table>
<thead>
<tr>
<th></th>
<th>Linear layer</th>
<th>Non-linear layer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gain</td>
<td></td>
</tr>
<tr>
<td>shuffling-tuples</td>
<td>$\eta$</td>
<td>$\eta$</td>
</tr>
<tr>
<td>shuffling-shares</td>
<td>$\eta^d$</td>
<td>$\eta$</td>
</tr>
<tr>
<td>shuffling-everything</td>
<td>$\left(\frac{d \cdot \eta}{d}\right)$</td>
<td>$d \cdot \eta$</td>
</tr>
</tbody>
</table>

Table: Summary of the shuffling + masking combinations.

→ Next focus on shuffling-shares.
Contents

Introduction

Linear layers

Non-linear layers

Perf. vs security
Time versus security for shuffled ISW: open questions

Bitslice masking:

- Favors large \#ANDs.
- Profits from parallelism.
- Randomness usage:

\[
\#\text{AND} \cdot \frac{d \cdot (d - 1)}{2}
\]
Time versus security for shuffled ISW: open questions

Bitslice masking:
- Favors large \#ANDs.
- Profits from parallelism.
- Randomness usage:
  \[
  \#\text{AND} \cdot \frac{d \cdot (d - 1)}{2}
  \]

Shuffling:
- Favors large \#ANDs.
- Profits from serialization.
- Randomness usage:
  \[
  d^2 \cdot \eta \cdot \log_2 \eta
  \]
Time versus security for shuffled ISW: open questions

Bitslice masking:
- Favors large \#ANDs.
- Profits from parallelism.
- Randomness usage:
  \[ \#\text{AND} \cdot \frac{d \cdot (d - 1)}{2} \]

Shuffling:
- Favors large \#ANDs.
- Profits from serialization.
- Randomness usage:
  \[ d^2 \cdot \eta \cdot \log_2 \eta \]

Challenges when protecting ISW:
- Should we favor parallelism or serialization.
- Does it depend on the platform?
- Does it depend on the primitive to protect?
Time versus security for shuffled ISW: design space  

Option 1:  
▶ Only bitsliced ISW.  
▶ 32 bits per reg (full para.).

Option 2:  
▶ Shuffled bitsliced ISW.  
▶ 32 bits per reg (full para.).

Option 3:  
▶ Shuffled bitsliced ISW.  
▶ 16 bits per reg (inc. ser.).
Introduction  Linear layers  Non-linear layers  Perf. vs security

Time versus security for shuffled ISW: design space  (#AND = 64)

Option 1:

- Only bitsliced ISW.
- 32 bits per reg (full para.).
Time versus security for shuffled ISW: design space (\#AND = 64)

Option 1:
- Only bitsliced ISW.
- 32 bits per reg (full para.).

Option 2:
- Shuffled bitsliced ISW.
- 32 bits per reg (full para.).
Time versus security for shuffled ISW: design space  (#\text{AND} = 64)

Option 1:
- Only bitsliced ISW.
- 32 bits per reg (full para.).

Option 2:
- Shuffled bitsliced ISW.
- 32 bits per reg (full para.).

Option 3:
- Shuffled bitsliced ISW.
- 16 bits per reg (inc. ser.).
Time versus security: experimental results

![Graph showing the relationship between time and security for different RNG options.](image-url)
Time versus security: experimental results

Opt 1: mask. only
Time versus security: experimental results

Cheap RNG

Expensive RNG

Opt 1: mask. only
Opt 2: mask. & shuffl.
Time versus security: experimental results

Opt 1: mask. only
Opt 2: mask. & shuffl.
Opt 3: larger perm.
Time versus security: experimental results

Opt 1: mask. only
Opt 2: mask. & shuffl.
Opt 3: larger perm.

Take home:
Use fully the registers and then shuffle.
General conclusion for masking and shuffling combination

- When to favor shuffling + masking:
  ▶ large number of independent inputs AND expensive randomness.
  ▶ relatively low noise.

Thanks!

Masking or Masking + shuffling:
- masking is faster.
- masking + shuffling is faster.

https://github.com/uclcrypto/bitslice_masking_and_shuffling
General conclusion for masking and shuffling combination

When to favor shuffling + masking:
▶ large number of independent values
▶ expensive randomness $r$
▶ relatively low noise.

Thanks!

Masking or Masking + shuffling:
▶ masking is faster.
▶ masking + shuffling is faster.

https://github.com/uclcrypto/bitslice_masking_and_shuffling
General conclusion for masking and shuffling combination

Masking or Masking + shuffling:

- : masking is faster.
- : masking + shuffling is faster.

Figure: \#AND=128, \( N = 2^{64} \)
General conclusion for masking and shuffling combination

Masking or Masking + shuffling:

- ▶️: masking is faster.
- □: masking + shuffling is faster.

Figure: \#AND=256, \( N = 2^{64} \)
General conclusion for masking and shuffling combination

Masking or Masking + shuffling:

▶ black: masking is faster.
▶ white: masking + shuffling is faster.

Figure: \#AND=512, N = 2^{64}
General conclusion for masking and shuffling combination

Masking or Masking + shuffling:

▶ : masking is faster.
▶ : masking + shuffling is faster.

Figure: \#AND=1024, \( N = 2^{64} \)
General conclusion for masking and shuffling combination

When to favor shuffling + masking:
- large of independent $\#AND$.
- expensive randomness $r$.
- relatively low noise.

Masking or Masking + shuffling:
- •: masking is faster.
- □: masking + shuffling is faster.

Figure: $\#AND=1024$, $N = 2^{64}$
General conclusion for masking and shuffling combination

When to favor shuffling + masking:
- large of independent \( #AND \).
- expensive randomness \( r \).
- relatively low noise.

Thanks!

Masking or Masking + shuffling:

- ▶️: masking is faster.
- ▶️: masking + shuffling is faster.

Figure: \( #AND=1024, N = 2^{64} \)

https://github.com/uclcrypto/bitslice_masking_and_shuffling