Towards Globally Optimized Masking: From Low Randomness to Low Noise Rate

Gaëtan Cassiers  François-Xavier Standaert
CHES 2019
The threat of horizontal attacks

DPA
The threat of horizontal attacks

Horizontal attack
Masking a sensitive bit $x$:

$$x = x_0 \oplus \cdots \oplus x_{d-2} \oplus x_{d-1}$$

and compute only on sharing $(x_0, \ldots, x_{d-1})$. 

What about horizontal attacks? 

[BCPZ16]: Qualitative analysis and countermeasure.
Just mask it!

Masking a sensitive bit $x$:

$$x = x_0 \bigoplus \cdots \bigoplus x_{d-2} \bigoplus x_{d-1}$$

random

and compute only on sharing $(x_0, \ldots, x_{d-1})$.

Secure in the probing model at order $t$.

What about horizontal attacks?

[BCPZ16]: Qualitative analysis and countermeasure.
Contributions

Horizontal attacks against masking:
- Quantitative (heuristic-based) analysis
- Automated tool

Countermeasures:
- Improved masked multiplication gadget
Outline

Introduction

Masked gadgets

Local Random Probing Model (LRPM)

LRPM bounds & improved gadgets

Conclusion
(ISW03]-like) AND gadget: \( z = x \otimes y \)

\[
\begin{pmatrix}
    z_0 \\
    z_1 \\
    z_2
\end{pmatrix} = \begin{pmatrix}
    x_0 \otimes y_0 & \oplus & (x_0 \otimes y_1 \oplus r_0) & \oplus & (x_0 \otimes y_2 \oplus r_1) \\
    (x_1 \otimes y_0 \oplus r_0) & \oplus & x_1 \otimes y_1 & \oplus & (x_1 \otimes y_2 \oplus r_2) \\
    (x_2 \otimes y_0 \oplus r_1) & \oplus & (x_2 \otimes y_1 \oplus r_2) & \oplus & x_2 \otimes y_2
\end{pmatrix}
\]

\[
\begin{bmatrix}
    x_0 \\
    x_1 \\
    x_2
\end{bmatrix}, \begin{bmatrix}
    y_0 \\
    y_1 \\
    y_2
\end{bmatrix} \xrightarrow{\text{MatGen}} \begin{bmatrix}
    (x_{0,0}, y_{0,0}) & (x_{0,1}, y_{1,0}) & (x_{0,2}, y_{2,0}) \\
    (x_{1,0}, y_{0,1}) & (x_{1,1}, y_{1,1}) & (x_{1,2}, y_{2,1}) \\
    (x_{2,0}, y_{0,2}) & (x_{2,1}, y_{1,2}) & (x_{2,2}, y_{2,2})
\end{bmatrix} \ldots
\]

\[
\begin{array}{c}
\ldots \xrightarrow{\text{Product}} \begin{bmatrix}
    \alpha_{0,0} & \alpha_{0,1} & \alpha_{0,2} \\
    \alpha_{1,0} & \alpha_{1,1} & \alpha_{1,2} \\
    \alpha_{2,0} & \alpha_{2,1} & \alpha_{2,2}
\end{bmatrix} \\
\text{Compression} \xrightarrow{\oplus_j \alpha_{i,j} \oplus r_{i,j}} \begin{bmatrix}
    z_0 \\
    z_1 \\
    z_2
\end{bmatrix}
\end{array}
\]
Other Masked AND gates

\[
\begin{bmatrix}
  x_0 \\
  x_1 \\
  x_2
\end{bmatrix}, \begin{bmatrix}
  y_0 \\
  y_1 \\
  y_2
\end{bmatrix} \xrightarrow{\text{MatGen}} \begin{bmatrix}
  (x_0,0, y_0,0) \\
  (x_1,0, y_0,1) \\
  (x_2,0, y_0,2)
\end{bmatrix}, \begin{bmatrix}
  (x_0,1, y_1,0) \\
  (x_1,1, y_1,1) \\
  (x_2,1, y_1,2)
\end{bmatrix}, \begin{bmatrix}
  (x_0,2, y_2,0) \\
  (x_1,2, y_2,1) \\
  (x_2,2, y_2,2)
\end{bmatrix} \xrightarrow{\text{Prod.}} \begin{bmatrix}
  \alpha_{0,0} \\
  \alpha_{1,0} \\
  \alpha_{2,0}
\end{bmatrix}, \begin{bmatrix}
  \alpha_{0,1} \\
  \alpha_{1,1} \\
  \alpha_{2,1}
\end{bmatrix}, \begin{bmatrix}
  \alpha_{0,2} \\
  \alpha_{1,2} \\
  \alpha_{2,2}
\end{bmatrix} \xrightarrow{\text{Comp.}} \begin{bmatrix}
  z_0 \\
  z_1 \\
  z_2
\end{bmatrix}
\]

Compression

- Reduced randomness requirement [BBP+16].

Product

- Other security property for composition (PINI) [CS18].

MatGen

- Security against horizontal attacks [BCPZ16].
Outline

Introduction

Masked gadgets

Local Random Probing Model (LRPM)

LRPM bounds & improved gadgets

Conclusion
Factor graph & Belief propagation

\[ x = g(y) \quad x = x_1 \oplus x_2 \quad z = x_2 \otimes y \]

\( \mathcal{L} \): intrinsic information from leakage trace.

**Belief propagation** (BP): estimate \( x, y \) given information about \( x, y, x_1, x_2, z \).

Alternate propagation of distribution estimates (beliefs):

Variable nodes \( \leftrightarrow \) Function nodes

Basis for Soft-Analytical Side-Channel Attacks (SASCA).
Belief propagation (BP): estimate $x, y$ given information about $x, y, x_1, x_2, z$.

Alternate propagation of distribution estimates (beliefs):

Variable nodes $\iff$ Function nodes

Basis for Soft-Analytical Side-Channel Attacks (SASCA).
Belief propagation (BP): estimate $x, y$ given information about $x, y, x_1, x_2, z$.

Alternate propagation of distribution estimates (beliefs):

Variable nodes $\leftrightarrow$ Function nodes

Basis for Soft-Analytical Side-Channel Attacks (SASCA).
Local Random Probing Model (LRPM)

$\epsilon$-Random probing model
Observe for each variable $x$:

\[
\begin{cases}
  x & \text{with probability } \epsilon \\
  \perp & \text{with probability } 1 - \epsilon
\end{cases}
\]

Local random probing model [GGS18]:
Random probing model adversary using BP.

Computing bounds in the LRPM:

Adaptation of BP to estimate mutual information (MI) instead of distributions.

- Input: *noise level* as observation MI on manipulated variables.
- Result: *security level* as MI on sensitive target variables.
LRPM example: Multiplication gadget

\[ \begin{align*}
p_0 &= x_0 \otimes y_0 \\
p_1 &= x_0 \otimes y_1 \\
p_2 &= x_1 \otimes y_0 \\
p_3 &= x_1 \otimes y_1 \\
\alpha_0 &= p_1 \oplus r_0 \\
\alpha_1 &= p_2 \oplus r_0 \\
z_0 &= p_0 \oplus \alpha_0 \\
z_1 &= p_3 \oplus \alpha_1
\end{align*} \]

- Every operand/result computation leaks (sources of thick edges)
Outline

Introduction

Masked gadgets

Local Random Probing Model (LRPM)

LRPM bounds & improved gadgets

Conclusion
LRPM bound: [ISW03] masked AND

- Required noise increases with #shares
- More shares may be worse.

1/Noise (Obs. MI [bit])

1/Security (Target MI [bit])

- 2 shares
- 8 shares
- 32 shares

1/Noise (Obs. MI [bit])
LRPM bound: [ISW03] masked AND

- Required noise increases with \# shares
- More shares may be worse.

![Graph showing the relationship between noise and security for different number of shares.](image)
Reducing shares re-use: MatGen (I)

[ISW03]

- Simplest strategy
- Maximal efficiency
- Each input share used $d$ times
Reducing shares re-use: MatGen (II)

[BCPZ116]

- Add refreshing before shares multiplication
- Each input share used log $d$ times
Reducing shares re-use: MatGen (III)

This work: many cheap refreshings

- 2 refresh gadgets per layer
- Each input share used 3 times
AND implementations: Gadget comparison

\[ d = 16 \]

\[ \frac{1}{\text{Noise (Obs. MI [bit])}} \]

\[ \frac{1}{\text{Security (Target MI [bit])}} \]

\[ 2^{-76} \]

\[ 2^{-7} \]

\[ 2^{-6} \]

\[ 2^{-5} \]

\[ 2^{-19} \]

\[ 2^{-38} \]

\[ 2^{-57} \]

\[ 2^0 \]

\[ \text{[ISW03]} \]

\[ \text{[BCPZ16]} \]

\[ \text{Ours} \]
AND implementation: [ISW03]

- $d = 2, \ldots, 32$
- Noise rate: $1/d$

The graph shows the relationship between $1/\text{Noise (Obs. MI [bit])}$ and $1/\text{Security (Target MI [bit])}$ for different values of $d$. The line labels indicate the noise rate $1/d$. The graph includes annotations for $2^{-8}$, $2^{-7}$, $2^{-6}$, $2^{-5}$, and $2^{-4}$ on both axes.
AND implementations: [BCPZ16]

- $d = 2, \ldots, 32$
- Noise rate: $1/\log(d)$
AND implementations: Ours

- $d = 2, \ldots, 32$
- Noise rate: 1
Cost: 40 bit security

![Graph showing relative runtime cost vs. 1/Noise (Obs. MI [bit])](image)

- [ISW03]
- [BCPZ16]
- Ours

- d=6
- d=32
Cost: 60 bit security

Relative runtime cost

1/Noise (Obs. MI [bit])

[ISW03] [BCPZ16] Ours

d=9 d=32 d=32
Conclusion

Horizontal attacks:

- Qualitative $\rightarrow$ Automated quantitative analysis.
- New multiplication gadget: conjectured $O(1)$ noise rate (in $\mathbb{F}_2$).
- Randomness $+$ Computations vs Noise: implementer’s trade-off.

Not presented here:

- New composable (PINI) gadget reducing randomness.

Thank you!
Conclusion

Horizontal attacks:

- Qualitative $\rightarrow$ Automated quantitative analysis.
- New multiplication gadget: conjectured $O(1)$ noise rate (in $\mathbb{F}_2$).
- Randomness+Computations vs Noise: implementer’s trade-off.

Not presented here:

- New composable (PINI) gadget reducing randomness.
Conclusion

Horizontal attacks:

- Qualitative $\rightarrow$ Automated quantitative analysis.
- New multiplication gadget: conjectured $O(1)$ noise rate (in $\mathbb{F}_2$).
- Randomness + Computations vs Noise: implementer’s trade-off.

Not presented here:

- New composable (PINI) gadget reducing randomness.

Thank you!