Towards Leakage Simulators that Withstand the Correlation Distinguisher

P. Pessl, F.-X. Standaert, S. Mangard, F. Durvaux
IAIK TU Graz (Austria), UCL Crypto Group (Belgium)

ASIACRYPT rump session, December 2014
Background

- Split & Concatenate Simulator (CRYPTO 2013)

\[
L(x, k, y) \approx L(x, \tilde{k}, y^*) || L(x^*, \tilde{k}, y)
\]
Background

- Split & Concatenate Simulator (CRYPTO 2013)

\[ L(x, k, y) \approx L(x, \tilde{k}, y^*) \| L(x^*, \tilde{k}, y) \]

- Longo Galea et al (ASIACRYPT 2014): \( \exists \) correlation between samples within real traces (e.g. \( \rho > 0.5 \)) … that are significantly reduced in simulated ones \( \Rightarrow \) Allows distinguishing!
Background

- **Split & Concatenate Simulator (CRYPTO 2013)**

\[ L(x, k, y) \approx \begin{cases} L(x, \tilde{k}, y^*) \parallel L(x^*, \tilde{k}, y) \end{cases} \]

- **Longo Galea et al (ASIACRYPT 2014):** \( \exists \) correlation between samples within real traces (e.g. \( \rho > 0.5 \)) … that are significantly reduced in simulated ones ⇒ Allows distinguishing!

- **Proposed solution:** very noisy implementations, *but it scales badly*: noise arbitrarily reduced with averaging
Background

- Split & Concatenate Simulator (CRYPTO 2013)
  \[ L(x, k, y) \approx L(x, \tilde{k}, y^*) || L(x^*, \tilde{k}, y) \]

- Longo Galea et al (ASIACRYPT 2014): \( \exists \) correlation between samples within real traces (e.g. \( \rho > 0.5 \)) … that are significantly reduced in simulated ones ⇒ Allows distinguishing!

- Proposed solution: very noisy implementations, but it scales badly: noise arbitrarily reduced with averaging

Can we do better?
Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho(x, Sbox(x)) \ll 0.5$
Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho(x, \text{Sbox}(x)) \ll 0.5$
- Physical then $\Rightarrow$ let’s use a simple physical model
Origin of the intra-trace correlation

• Algorithmic? Unlikely: $\rho(x, \text{Sbox}(x)) \ll 0.5$
• Physical then $\Rightarrow$ let’s use a simple physical model

\[ L(x, k, y) = \delta(x, k, y) + N \]

\( \text{signal} \quad \text{noise} \)
Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho(x, \text{Sbox}(x)) \ll 0.5$
- Physical then $\Rightarrow$ let’s use a simple physical model

$$L(x, k, y) = \delta(x, k, y) + N$$

$\Rightarrow$ Does the correlation come from signal or noise?
Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho(x, \text{Sbox}(x)) \ll 0.5$
- Physical then $\Rightarrow$ let’s use a simple physical model

$$L(x, k, y) = \delta(x, k, y) + N$$

signal noise

$\Rightarrow$ Does the correlation come from signal or noise?

- In particular for large parallel implementations (since we know 8-bit AES implementations can be broken in one trace anyway – see SASCA paper)
Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)
Repeating experiments with a 65nm ASIC

• Intra-trace correlation (real traces, sample 500)

• Same, with simulated traces $L(x, \tilde{k}, y^*) || L(x^*, \tilde{k}, y)$
Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)

- Same, with simulated traces $L(x, \tilde{k}, y^*) || L(x^*, \tilde{k}, y)$

- & fake simulated traces $\delta(x, k, y) + N_1 || \delta(x, k, y) + N_2$
Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)

- Same, with simulated traces $L(x, k, y)$

- & fake simulated traces $\delta(x, k, y) + N_1 || \delta(x, k, y) + N_2$
A first improvement

- Sliding simulator

\[ L(x, \tilde{k}, y^*) \cdot \text{\ding{65}} + L(x^*, \tilde{k}, y) \cdot \text{\ding{65}} \]
A first improvement

- **Sliding simulator**
  \[ L(x, \tilde{k}, y^*) \cdot ⬇️ + L(x^*, \tilde{k}, y) \cdot ⬇️ \]

- **Real traces**
A first improvement

- Sliding simulator

\[ L(x, \tilde{k}, y^*) \cdot \begin{array}{c}
\text{\large \blacktriangleleft} \\
\end{array} + L(x^*, \tilde{k}, y) \cdot \begin{array}{c}
\text{\large \blacktriangleleft} \\
\end{array} \]

- Real traces

- Simulated traces
A first improvement

- Sliding simulator
  \[ L(x, \tilde{k}, y^*) \cdot \blacktriangleleft + L(x^*, \tilde{k}, y) \cdot \blacktriangleleft \]

- Real traces

- Simulated traces
  BETTER BUT NOT ENOUGH
The main idea: separate signal and noise

- Sliding signal + noise simulator

\[ \hat{\delta}(x, \tilde{k}, y^*) \cdot \blacktriangle + \hat{\delta}(x^*, \tilde{k}, y) \cdot \blacktriangle + N \]
The main idea: separate signal and noise

- Sliding signal + noise simulator

\[ \hat{\delta}(x, \tilde{k}, y^*) \cdot \ \text{avg. trace} \ + \ \hat{\delta}(x^*, \tilde{k}, y) \cdot \ \text{avg. trace} - \text{single trace} \ + N \]
The main idea: separate signal and noise

- Sliding signal + noise simulator
  \[ \delta(x, \bar{k}, y^*) \cdot \downarrow + \delta(x^*, \bar{k}, y) \cdot \downarrow + N \]
  avg. trace \quad \text{avg. trace – single trace}

- Real traces

![Graph showing cross-correlation over time](image)
The main idea: separate signal and noise

- Sliding signal + noise simulator
  
  \[ \hat{\delta}(x, \tilde{k}, y^*) \cdot \downarrow + \hat{\delta}(x^*, \tilde{k}, y) \cdot \downarrow + N \]

  avg. trace  \quad \text{avg. trace – single trace}

- Real traces

- Simulated traces
The main idea: separate signal and noise

- Sliding signal + noise simulator
  \[ \hat{\delta}(x, \tilde{k}, y^*) \cdot \triangleleft + \hat{\delta}(x^*, \tilde{k}, y) \cdot \triangleleft + N \]
  avg. trace

- Real traces
  \[ \text{LOOKS GOOD (obviously no noise-related correlation)} \]

- Simulated traces
Is it enough?

- Sliding S + N simulator prevents the $\rho$ distinguisher in contexts where noise-based correlation dominates
- (& the signal is hard to exploit/hybridize)
- **Achievable for certain large // implementations**
Is it enough?

• Sliding S + N simulator prevents the $\rho$ distinguisher in contexts where noise-based correlation dominates
  • (& the signal is hard to exploit/hybridize)
  • Achievable for certain large // implementations

• Work in progress. Further investigations are needed
  • Maintain the signal variance (modified because of the sum in the sliding simulator): easy!
  • Different settings, simulators, designs, …
Is it enough?

- Sliding S + N simulator prevents the $\rho$ distinguisher in contexts where noise-based correlation dominates
- ( & the signal is hard to exploit/hybridize)
- *Achievable for certain large // implementations*

- **Work in progress.** Further investigations are needed
  - Maintain the signal variance (modified because of the sum in the sliding simulator): easy!
  - Different settings, simulators, designs, …

*Reminder: simulatability is the only empirically verifiable leakage assumption we currently have!*
STAY TUNED

http://perso.uclouvain.be/fstandae/