Practical Evaluation of FSE 2016 Customized Encoding Countermeasure

Shivam Bhasin$^1$, Dirmanto Jap$^1$, Thomas Peyrin$^{1,2,3}$

$^1$Temasek Laboratories
$^2$School Of Physical and Mathematical Sciences
$^3$School of Computer Science and Engineering
Nanyang Technological University, Singapore

FSE 2018
Brugge, Belgium
1. Context

2. Hiding Countermeasure in Software

3. Practical Analysis

4. Conclusions
1 Context

2 Hiding Countermeasure in Software

3 Practical Analysis

4 Conclusions
Side-Channel Attacks (SCA)

Source: http://www.inmagine.com
Side-Channel Attacks (SCA)
**SCA Countermeasure: Masking**

**Basic Principle**

- ⇒ Randomization of the sensitive data\(^1\).
- Power consumption uncorrelated to data.

---

\(^1\)Coron et al, CHES 2000
SCA Countermeasure: Hiding

Dual Rail and Precharge Logic (DPL)

- ⇒ Data-Independent Power Consumption
- Duplication ⇒ Balanced Activity\(^2\)
- Two Phases ⇒ Constant Transitions.
- 0→01, 1→10, precharge→00, invalid→11.

\(^2\)Tiri et al, DATE 2004.
Hiding Countermeasure in Software
Hiding Countermeasure in Software

- Idea introduced by Hoogvorst et al in 2011\(^3\)
- Adopt DPL principle for data representation in software.
- Aimed to reduce (or remove) data dependence of power consumption. Both data and operations are adjusted to enable processing of encoded data.
- Two further proposals:
  - Balanced bit slicing, following DPL method\(^4\): 0\(\mapsto\)01, 1\(\mapsto\)10
  - Balanced Encoding\(^5\): \(b_3\overline{b}_3b_2\overline{b}_2b_1\overline{b}_1b_0\overline{b}_0\).
- In practice, both leak but reduce SNR.
- Shows additional fault resistance properties\(^6\).

\(^3\)Hoogvorst et al, COSADE 2011.
\(^4\)Rauzy et al., PROOFS 2014
\(^5\)Chen et al., CARDIS 2014
\(^6\)Breier et al, HOST 2016.
Why Does it Leaks?

- Device physics
- DPL assumes equal bit contribution/weight
- In reality, bits have unequal contribution
- Perfect HW/HD model are hard to realise
Why Does it Leaks?

Perfect Setting
Why Does it Leaks?

Real Setting
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
  - There is Wisdom in Harnessing the Strengths of Your Enemy
  - Profile actual bit weights ($\beta$) from the device
  - Compute encoding from the bit weights to minimise bias
  - Longer encodings (vs 2 bits for DPL)
  - Previously demonstrated to protect $S$-box look-up
  - Vary from one device copy to another

\textsuperscript{7} Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
- *There is Wisdom in Harnessing the Strengths of Your Enemy*
  - Profile actual bit weights ($\beta$) from the device
  - Compute encoding from the bit weights to minimise bias
  - Longer encodings (vs 2 bits for DPL)
  - Previously demonstrated to protect Sbox look-up
  - Vary from one device copy to another

\textsuperscript{7}Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\(^7\)
- *There is Wisdom in Harnessing the Strengths of Your Enemy*
- Profile actual bit weights (\(\beta\)) from the device
  - Compute encoding from the bit weights to minimise bias
  - Longer encodings (vs 2 bits for DPL)
  - Previously demonstrated to protect *Sbox look-up*
  - Vary from one device copy to another

\(^7\) Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
- *There is Wisdom in Harnessing the Strengths of Your Enemy*
- Profile actual bit weights (\(\beta\)) from the device
- Compute encoding from the bit weights to minimise bias
  - Longer encodings (vs 2 bits for DPL)
  - Previously demonstrated to protect \textit{Sbox look-up}
  - Vary from one device copy to another

\textsuperscript{7}Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
- *There is Wisdom in Harnessing the Strengths of Your Enemy*
- Profile actual bit weights ($\beta$) from the device
- Compute encoding from the bit weights to minimise bias
- Longer encodings (vs 2 bits for DPL)
  - Previously demonstrated to protect \textit{Sbox look-up}
  - Vary from one device copy to another

\textsuperscript{7}Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
- \textit{There is Wisdom in Harnessing the Strengths of Your Enemy}
- Profile actual bit weights ($\beta$) from the device
- Compute encoding from the bit weights to minimise bias
- Longer encodings (vs 2 bits for DPL)
- Previously demonstrated to protect \textit{Sbox look-up}
- Vary from one device copy to another

\textsuperscript{7}Maghrebi et al, FSE 2016.
Customised Encoding Countermeasure

- Proposed by Maghrebi et al.\textsuperscript{7}
- \textit{There is Wisdom in Harnessing the Strengths of Your Enemy}
- Profile actual bit weights ($\beta$) from the device
- Compute encoding from the bit weights to minimise bias
- Longer encodings (vs 2 bits for DPL)
- Previously demonstrated to protect \textit{Sbox look-up}
- Vary from one device copy to another

\textsuperscript{7} Maghrebi et al, FSE 2016.
Simulated Analysis of Customised Encoding

- Derived values from real EM measurements
- AES on 8-bit AVR microcontroller
- Profile for $\beta$ and noise variances
- Variance of $\beta \in [0.2, 0.8]$  
- Variance of noise $\in [5.5, 6.8]$  
- Use TVLA$^8$ based analysis
- Considered leaking data-dependant information if $t / \notin [-4.5, 4.5]$

---

$^8$Goodwill et al, NIAT 2011.
Simulated Analysis of Customised Encoding

Figure: TVLA results for unprotected and countermeasure (5 to 10 bits encoding and software dual-rail (SW-DR)) with different $\beta$ variances.
Simulated Analysis of Customised Encoding

(a) 8-bit encoding

(b) 10-bit encoding

Figure: TVLA results for 8 to 10-bit encoding schemes with different noise levels

Longer encoding helps
Context

Hiding Countermeasure in Software

Practical Analysis

Conclusions
Building Customised Encoding

(a) Region of Interest

(b) CPA results

Figure: Feature selection for $\beta$.

- EM measurement on AVR for AES Sbox (LDR+STR)
- $\beta$ averaged over clock of highest correlation
- Two encodings $a_1$ and $a_2$ derived
- Used to implement lightweight SKINNY
Impact of Changing the Register

Figure: TVLA on encoding a1

- Implementing whole cipher with one instruction and register can be difficult
- Protecting one instruction and register is possible
- Encoding must be updated with change in register
Impact of Measurement Method

(a) TVLA EM  
(b) TVLA Power  
(c) $\beta$ EM vs power

Figure: Leakage profiling comparison: EM vs Power. (c) The $\beta$ coefficients obtained from EM and power under the same setup.

- Similar observations for different EM positions, time samples.
- Updating/Converting encoding can be costly and leak
Longer & Higher Order Encoding

- Tested longer encodings with 32-bit ARM microcontroller
- Limited to 10 bit encoding due to memory size
- Also tested higher order (HO) encoding taking not only individual $\beta$ but their coupling affect to arrive at a more precise encoding.
Longer & Higher Order Encoding

(a) Key rank unprotected
(b) Key rank customized encoding

(c) Key Rank HO customized encoding
(d) CPA HO customised encoding
1 Context

2 Hiding Countermeasure in Software

3 Practical Analysis

4 Conclusions
Conclusion

- Practically evaluated Customised encoding countermeasure
- Shown sound in simulations
- In practice, temporal and spatial variance of $\beta$ prevents effective encoding
- Hard to obtain a generic encoding
- Implementing a full cipher was difficult
- Several test cases highlighted on two different microcontrollers
- $\beta$ based estimation works well for attacks but its relation with device physics is not clear
- Studying it will help develop strong countermeasures
Thank you!
Any questions?