Rejection Sampling Schemes for Extracting Uniform Distribution from Biased PUFs

Rei Ueno, Kohei Kazumori, and Naofumi Homma
Tohoku University
Background

- Physically unclonable functions (PUFs) play essential role for constructing secure and trustable systems
  - Generate hardware-intrinsic random number like fingerprint
  - Exploit process variations for physical unclonability and tamper evidence

- Major applications of PUF
  - Entity authentication (Strong PUF)
  - Cryptographic key generation (Weak PUF)

Even for same input and same circuit construction, PUF responses vary due to process variation (i.e., $R_1 \neq R_2 \neq \cdots$)
PUF-based key generation

- Fuzzy extractor (FE) is commonly used for reconstructing enrolled key from noisy PUF response

- Helper data is stored in common nonvolatile memory (NVM)
  - NVM is usually non-tamper resistant, and helper data is considered public
  - We should consider conditional entropy for key generation
    - A $\sigma$-bit key generation is realized only if $\mathbb{H}(S|W) > \sigma$
Problem of PUF bias: Entropy leakage

- If PUF response is unbiased, $\mathbb{H}(S|W) = \mathbb{H}(S)$ (i.e., seed length)
- But $\mathbb{H}(S|W)$ significantly decreases with PUF bias increase
  - $\mathbb{H}(S|W) = \mathbb{H}(S) - \mathbb{I}(S; W)$ Entropy leakage
  - If PUF is biased, random seed should be set longer than $\sigma$ such that $\mathbb{H}(S|W) > \sigma$
    - But required PUF size rapidly grows with PUF bias, especially when $p_1 > 0.58$

**Channel diagram of FE [HO17]**

**PUF size required for reliable 128-bit key generation**

(Values are from [DGV+16])

<table>
<thead>
<tr>
<th>PUF bias $p_1$</th>
<th>0.54</th>
<th>0.58</th>
<th>0.62</th>
<th>0.66</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit-error rate</td>
<td>0.100</td>
<td>0.098</td>
<td>0.096</td>
<td>0.092</td>
</tr>
<tr>
<td>PUF size</td>
<td>1,530</td>
<td>2,550</td>
<td>5,100</td>
<td>13,005</td>
</tr>
</tbody>
</table>
Debiasing

• Extract unbiased bit string from biased PUF response
  • Realize secure key generation even from PUFs with nonnegligible biases
  • Efficiency has been evaluated through PUF size required for reliable 128-bit key gen.

• Example of debiasing: von Neumann corrector (VNC)
  • Values of 1 and 0 are extracted with an identical probability of \( p_1 p_0 \)
  • Debiasing data \( d \) is used for reproducing \( z \) at reconstruction

PUF response \( x: \)

\[
\begin{array}{cccccccccc}
1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
& & & & & & & & & \\
\end{array}
\]

Debiased data \( z: \)

\[
\begin{array}{cccccc}
1 & 0 & & & & \\
\end{array}
\]

Debiasing data \( d: \)

\[
\begin{array}{cccc}
1 & 1 & 0 & 0 \\
\end{array}
\]
Conventional debiasing-based FEs

- Various debiasing-based FEs have been developed for improving efficiency
  - Efficient FE reduces PUF and NVM sizes
  - How far can we go?

- [YD10] Index-based syndrome (IBS)
- [LSHT10] First von Neumann corrector (VNC)-based debiasing
- [HMSS12] Generalized IBS
- [KLRW14] Report on entropy loss in PUF-based key generation
- [MLSW15] VNC-based FE, first explicit solution for secure key generation from biased PUFs
- [DGV+16] Tight bounds of min-entropy loss, motivation for debiasing
- [SUHA17] Ternary VNC-based FEs
- [S17] Trivial debiasing
- [AWSO17] Maskless debiasing (MD)
- [HO17] Coset coding (CC)-based FE, FE is modeled as wire-tap channel
- [USH19] Biased masking (BM)-based FE
- [KW19] Selection and balancing schemes for SRAM PUF
This work

• **Acceptance-or-Rejection (AR)-based FE**: New debiasing scheme based on rejection sampling and FE construction
  • Extract uniform distribution with highest efficiency among conventional FEs
  • Implemented with solely an RNG at enrollment, and no critical additional operation is required at reconstruction performed on client device
  • First FE which can tolerate local biases depending on cell addresses (for example, found in some SRAM PUFs)
  • Extended to ternary PUF response for improved efficiency (see our paper)

• **Performance of proposed FE** is evaluated through simulation of 128-bit key generation in comparison with conventional FEs
  • AR-based FE achieves smallest PUF and/or NVM sizes (i.e., hardware cost) for various PUFs
  • At most 55% and 72% smaller PUF and/or NVM sizes than counterparts
Bias models

• Global bias model
  • All bits in PUF response have an identical bias of $p_1$ (with corresponding $p_0$)
  • All conventional debiasing scheme employed global bias model

• Cell-wise bias model (or local bias model)
  • Each bit has unique bias depending on cell address $i$
  • Expected value of biases are considered equal to global bias (i.e., $E_i[p_{1,i}] = p_1$)

Typical example of cell-wise-based PUF
Rejection sampling

• Method for deriving target distribution from proposal one
  • Target distribution: Distribution which is needed, but not directly available
  • Proposal distribution: Easily available distribution

- Target distribution: Uniform distribution
- Proposal distribution: PUF response (i.e., \( p_{1,i} \)-biased Bernoulli distribution)

Overview of rejection sampling

Step (1): Obtain sample \( a \) from \( p_{\text{prop}}(x) \)

Step (2): Draw random number \( b \) from \([0, p_{\text{prop}}(a)]\)

Step (3): Accept the sample if \( b < p_{\text{tar}}(a) \); otherwise, reject it

Application to PUF debiasing

• Target distribution: Uniform distribution
• Proposal distribution: PUF response (i.e., \( p_{1,i} \)-biased Bernoulli distribution)
Extraction of uniform distribution from biased PUFs

- Key idea: Bit-wise rejection sampling
  - Rejection sampling is applied to \( i \)-th cell with biases \( p_{1,i} \), \( p_{0,i} \) for all \( i \)
  - Expected length of debiased bit string is longer than conventional schemes
Proposed scheme: AR-based FE

- Reproducible rejection sampling (RRS) and accepted cell extraction (ACE) operations are applied to PUF response

<table>
<thead>
<tr>
<th>Enrollment of AR-based FE</th>
<th>Reconstruction of AR-based FE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUF ( x ) \rightarrow \text{RRS} ( d ) ACL data</td>
<td>PUF ( x' ) ACL data ( d \rightarrow \text{ACE} ( u' ) Helper data</td>
</tr>
<tr>
<td>\text{RNG} ( s ) \rightarrow \text{ECC encode} ( c ) Helper data</td>
<td>Helper data ( w \rightarrow \text{ECC decode} ( s ) KDF ( k ) key</td>
</tr>
<tr>
<td>\text{KDF} ( k ) key</td>
<td>ECC ( s ) KDF ( k ) key</td>
</tr>
</tbody>
</table>

- RRS operation generates debiased bit string and accepted cell location (ACL) data \( d \)
  - Naïve rejection sampling is not reproducible
  - ACL data enables us to reproduce debiased bits at ACL at reconstruction
  - We proved there is no entropy leakage from pair of helper and ACL data
**RRS and ACE operations—Implementation**

- RRS operation performs rejection sampling with reproducibility
  - First generate ACL data $d$, and then extract debiased bit string
  - Implemented using an RNG and bit-parallel operations in enrollment server

Step (0): Generate bit-string $h$, where $i$-th bit is Boolean value of $p_{1,i} \geq p_{0,i}$

Step (1): Take bit-parallel XOR of $x$ and $h$ (as $y$)

Step (2): Generate random number $r$ ($i$-th bit has a bias of $\min(p_{1,i}, p_{0,i})/\max(p_{1,i}, p_{0,i})$)

Step (3): Generate ACL data as $d = h \vee r$

Step (4): Obtain debiased bit string as extraction of $x_i$ with $d_i = 1$

- ACE operation extracts bit value of cells indicated by ACL data
  - No additional computation is required in reconstruction
AR-based FE—Features

• Security
  • No entropy leakage, and $\sigma$-bit random seed realizes $\sigma$-bit key generation

• Efficiency
  • Retained entropy via debiasing is given by $2mp_0$ (for $p_1 \geq p_0$) from $m$-bit PUF
  VNC [MLSW15]: $2mp_1p_0$, (Simplest one), MD [AWSO17]: $m/\mu$ ($\mu \geq 3$ for most cases), and TD [S17]: $2mp_0 - 2$

• Reliability
  • AR-based FE may fail enrollment if length of extracted bit string is insufficient
    • PUF size should be determined such that enrollment failure rate is smaller than threshold
    • Enrollment failure rate is feasibly calculated similarly to VNC-based FEs
  • RRS and ACE operations have no impact on bit-error rate of extracted bits
    • ECC can be designed in the same way as conventional FEs

• Implementation aspects
  • RNG and bit-parallel operation at enrollment are required as main overhead
  • Reconstruction require no additional computationally-critical operations
Performance evaluation

• Simulate 128-bit key generation to evaluate PUF and NVM sizes (i.e., hardware cost) for various biases and bit-error rates
  • PUF bias: 0.58—0.90
  • Bit-error rate: 0.025—0.100
  • ECC: BCH-repetition concatenate code
    • BCH codes with length of 7, 15, 31, 63, 127, and 255 are considered
  • Enrollment and reconstruction failure rates are set less than $10^{-6}$
  • Compared to VNC-, MD-, and BM-based FEs herein [MLSW15, AWSO17, USH19]
    • See our paper for comparison with other conventional FEs

Evaluation result

- AR-based FE achieves highest efficiency for most biases and bit-error rates
  - At most 55% smaller PUF size
  - NVM size is basically consistent with PUF size
Concluding remarks

• We present AR-based FE which extracts uniform distribution from biased PUFs based on rejection sampling
  • Implemented using RNG and bit-parallel operations on enrollment server
  • Client device with PUF requires no computational overhead
  • First debiasing scheme applicable to PUFs with local biases
  • Simulation of 128-bit key generation shows that AR-based FE has higher efficiency for most biases and bit-error rates than conventional FEs, and achieves at most 55% and/or 72% smaller PUF and NVM sizes respectively
  • Extended to ternary PUF response for improved efficiency (see our paper)
    • More efficient for many PUFs than counterparts (i.e., ternary VNC-based FEs and C-IBS)

• Future works
  • Real-world implementation and evaluation of key generation system based on AR-based FE
  • Extension of AR-based FE for secure reuse of PUF