Demo: Practical Cryptographically-Secure PUFs based on Learning Parity with Noise

Chenglu Jin*, Charles Herder†, Ling Ren†, Phuong Ha Nguyen*,
Benjamin Fuller*, Srinivas Devadas† and Marten van Dijk*
*University of Connecticut
Email: {chenglu.jin,phuong_ha.nguyen,benjamin.fuller,marten.van_dijk}@uconn.edu
†Massachusetts Institute of Technology
Email: chherder@gmail.com, {renling,devadas}@mit.edu

I. Project Description

Herder et al. designed a new computational fuzzy extractor and physical unclonable function (PUF) challenge-response protocol based on the Learning Parity with Noise (LPN) problem [1]. The protocol requires no storage on the PUF and can correct for significant measurement noise. However, Herder et al. did not implement their protocol. In this work, we give the first implementation of a challenge response protocol based on computational fuzzy extractors. Our construction is a simplified version of the design of Herder et al. and builds on a ring oscillator PUF. Our simplifications allow for a dramatic reduction in area by making a mild assumption on PUF output bits. Fig.1 depicts how a challenge response pair is generated by this PUF, and Fig.2 shows how the correct response can be regenerated by this PUF given a valid challenge. The security of this design can be reduced to a computational hardness assumption. Its self correction property uses confidence information which measures the reliability of each POK output bit. As shown in Fig1 and 2, this work requires a system level design which contains Ring Oscillator POK, Matrix-vector multiplier, Processor, TRNG and Hash function.

This work has not been published yet.

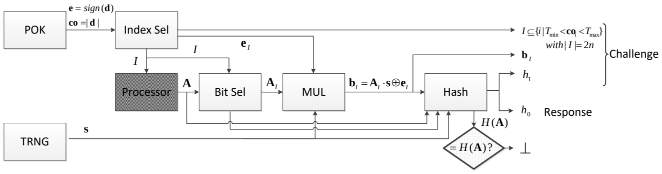
POK
$\mathbf{e} = sign(\mathbf{d})$
$\mathbf{co} = |\mathbf{d}|$
Index Sel
I
I
$\mathbf{e}_I$
Processor
$\mathbf{A}$
Bit Sel
$\mathbf{A}_I$
MUL
$\mathbf{b}_I = \mathbf{A}_I \cdot \mathbf{s} \oplus \mathbf{e}_I$
Hash
TRNG
$\mathbf{s}$
$H(\mathbf{A})$
$= H(\mathbf{A})?$
$\perp$
$I \subseteq \{i \mid T_{\min} < \mathbf{co}_i < T_{\max}\}$ $with |I| = 2n$
$\mathbf{b}_I$
h_1
Challenge
h_0
Response

Fig. 1. Gen: Gen produces a challenge response pairs (c, r), where $c = (I, \mathbf{b}_I, h_1)$ with $h_1 = H(\mathbf{A}_I, \mathbf{b}_I, s, 1)$ and $r = h_0$ with $h_0 = H(\mathbf{A}_I, \mathbf{b}_I, s, 0)$

II. Hardware Demo

For this demo, we only need one computer and one FPGA board. We will take both of them to the conference, so no extra equipment is needed from the organizer.

As it is mentioned in the description, one remarkable feature of this work is that this PUF construction can self-correct a large amount of measurement noise, so we will target on this feature in the demo.

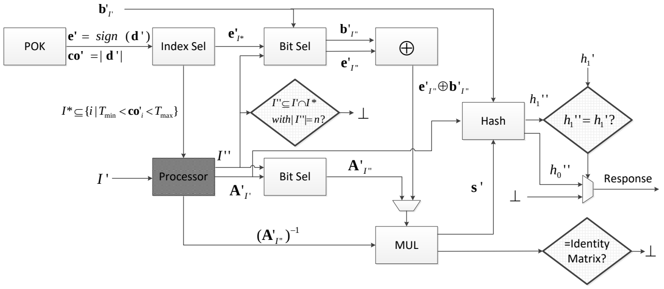

Fig. 2. Ver: Ver takes input $c = (I', \mathbf{b}'_I, h'_1)$, and either outputs an exception symbol $\perp$, or outputs the response $r = h''_0$. Notice that $\mathbf{e}'_{I''} \oplus \mathbf{b}'_{I''}$ is continuously fed into the multiplier with a small amount of bit flips (less than t bits).

Essentially, the observables for the audience would be the error rate of the raw POK output bits and how we correct these errors in the regeneration of responses.

In the poster which accompanies this demo, we will explain the internal structure of this design as shown in Fig.1 and 2.

References

[1] C. Herder, L. Ren, M. van Dijk, M. M. Yu, and S. Devadas, "Trapdoor Computational Fuzzy Extractors and Stateless Cryptographically-Secure Physical Unclonable Functions," *IEEE TDSC*, 2016.