Physically Unclonable Function: an Important Hardware Security Primitive

Rajat Subhra Chakraborty
Department of Computer Science and Engineering
Indian Institute of Technology
Kharagpur, West Bengal, INDIA - 721302
E-mail: rschakraborty@gmail.com

3rd NKN Workshop 2014
Outline

- PUF Fundamentals
  - Concept of PUF
  - PUF Quality Metrics
  - Applications of PUF in Security

- Example PUF Designs
  - Arbiter PUF (APUF)
  - Ring Oscillator PUFs (ROPUF)
  - Challenges in PUF Design

- Attacks on PUFs
  - Types of Attacks
  - Attack Example: Modeling Attack on APUF
SEAL-Lab (Secured Embedded Architecture Laboratory)

**Goal:**
- To design and analyze secured systems.
- To develop formal metrics for security.
- To develop design-for-security methodologies.

**Cryptography**
- Design and Analysis of Block Ciphers
- Boolean Functions
- Light Weight Cryptography
- Algebraic Cryptanalysis
- Application of Cellular Automaton to Cryptography
- Low Cost Protocol Design

**Crypto-Engineering**
- VLSI of Cryptographic algorithms
- Side Channel Analysis:
  - Power Attacks
  - Cache Attacks
- Fault Attacks
- Design-for-Testability for crypto devices
- Hardware Trojans, Watermarking

**Improved Attack on AES**
- Fig: Fault Attack on AES (needs 1 byte arbitrary fault to reduce AES-128 key to 256 possibilities)

**Research Highlights:**
- Strongest Fault Attack on AES (Advanced Encryption Standard), jointwork with Mike Tunstall, University of Bristol (UK) (ref. Africacrypt '09, WISTP '11)
- Cache Timing Attacks on CLEFIA cryptosystem (ref. Indocrypt '09, CTRSA '11)
- Hardware Prefetchers are responsible for leakage w.r.t. Cache attacks (HASP co-held with MICRO 12)
- Fastest characteristic-2 Elliptic Curve Crypto (ECC) processor on FPGAs (IEEE TVLSI), CHES 12
- Compact Cryptographically Robust Large Boolean Function (IEEE TC '11)

**Industry Partners:**
- NTT Labs (Japan), General Motors, Intel, DRDO (India)
- University Partners:
  - Bristol Univ. (UK), NYU-Poly (USA), Telecom Paris (France), K.U. Leuven (Belgium)

**Email:** debdeep@ece.iitd.ac.in

**Secured Embedded Architecture Laboratory (SEAL)**

**Invited Talks/Tutorials:**
- Darmstadt University (Germany), IEEE WIFS (Brazil), IEEE VLSI: D (India), ATS (India), JSIS (India), NTT Labs (Japan), NKC (UAE)

**International events hosted:**
- Organized first conference on Security Aspects in Information Technology in India, in association with the International Association of Cryptologic Research (IACR).
- Organized first workshop on Crypto-Hardware and Engineering in India jointly with Bristol Univ. and K.U.Leuven
Physically Unclonable Function (PUF)?

- Fingerprint Generator for Devices

- A challenge-response mechanism in which the mapping between an applied input ("challenge") and the corresponding observed output ("response") is dependent on the complex and variable nature of a physical material

- The challenge-response mapping is unclonable (ideally) and instance-specific

n-bit Challenge(C) \rightarrow PUF \rightarrow n-bit Response (R)
Silicon PUFs

- We are interested in PUF circuits, i.e. Silicon PUFs
- The dominant device for IC design is MOSFET
- Silicon PUFs utilize the unavoidable and unpredictable manufacturing process variation effects of modern deep-submicron MOSFET devices
- Usually, from CMOS circuit design perspective, process variation is a challenge, but is useful for PUF design
- Impact of process variation becomes more pronounced at advanced technology nodes
Quality Metrics for PUF

Uniqueness

\[ r_1 \neq r_2 \neq r_3 \]

PUF 1 \quad PUF 2 \quad PUF 3

Devices

C

Reliability

\[ r_1 = r_2 = r_3 \]

PUF 1 \quad PUF 1 \quad PUF 1

Time

C

- Other important properties: unpredictability and tamper-evidence
Why are PUFs Important?

Security without PUF

- Trusted party embeds and tests secret keys in a secure location (NVM)
- EEPROM adds additional complexity to manufacturing
- Adversaries may physically extract secret key from non-volatile memory

Security with PUF

- **Intrinsic** properties of devices are used to generate secret key.
- Key never leaves the IC’s cryptographic boundary, nor be stored in a non-volatile memory
- Key is deleted after usage in de- or encryption process
PUF in Use: Low-cost HW Authentication

- Protect against IC/FPGA substitution and counterfeits without using cryptographic operations

<table>
<thead>
<tr>
<th>Challenge</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001010</td>
<td>010101</td>
</tr>
<tr>
<td>1011000</td>
<td>101101</td>
</tr>
<tr>
<td>0111001</td>
<td>000110</td>
</tr>
</tbody>
</table>

Database for Device A

Is this the authentic Device A?
PUF response is used as a random seed to a private/public key generation algorithm
No secret needs to be handled by a manufacturer
A device generates a key pair on-chip, and outputs a public key
The public key can be endorsed at any time
- A randomized 3-round Luby-Rackoff cipher.
- Round functions are replaced PUF instances.
- This is a *keyless* cipher

[ Armknecht et al., ASIACRYPT 2009 ]
Composed of \( n \) two-port switching stages, for an \( n \)-bit challenge size

\( n \)-bit challenge \( \Rightarrow \) \( 2^n \) possible paths

Unique path selected by a challenge

Accumulated delay at the end of the path is compared by an arbiter circuit (usually, an edge-triggered D flip-flop)

Arbiter gives 1-bit decision

Advantages: Simple structure, low hardware overhead (each stage is two 2:1 MUXes)

Disadvantage: susceptible to modeling attacks
An $n$-bit applied challenge selects two different ROs from a bank of $2^n$ ROs.

Process variation implies ROs have different oscillation frequencies.

Compare frequencies of two oscillators using counters.

Comparator gives decision.

Advantage: Difficult to model

Disadvantage: Exponential hardware requirement.
### PUF Example 3: SRAM PUF

- Power-up initial value of SRAM cell can be used response, cell address is the challenge
- SRAM fabrication compatible with digital logic process in regular ICs
- FPGA implementation of SRAM PUF is very difficult (since SRAM modules are cleared by default on power-up)
Challenges in PUF Design

- Traditional CAD Tool based IC design flow is either inapplicable or infeasible to design PUFs
- **Reasons:**
  - Accurate simulation of process variation is difficult
  - Design, especially interconnect routing has to be carefully controlled to eliminate design bias (design bias adversely affects statistical quality of PUFs)
  - FPGA implementation of SRAM PUF is very difficult (since SRAM modules are cleared by default on power-up)
PUF Attack Overview

- Four paths leading to a PUF cloning attack
- Creating a physical clone of the PUF is considered infeasible
- The creation of a mathematical clone requires that the raw PUF response(s)
- Non-invasive attack methods using side channel analysis on the PUF
- Invasive attack involving mechanical probing of \( r' \)
- Attackers with access to contactless probing equipment can use a semi-invasive methodology to obtain the data of interest
Security Notion

A PUF P with \( n \)-bit challenge and \( m \)-bit response is considered as secure if it satisfies the following conditions:

1. No algorithm to predict the response \( R \) produced by an arbitrary PUF instance when an arbitrary challenge with probability of success greater than \( 2^{-m} \).
2. No algorithm to predict the response \( R \) for an arbitrary challenge with high probability of success, with sub-exponential time and space complexity.
3. No algorithm to predict the response \( R \) for an arbitrary challenge with high probability of success, with sub-exponential data complexity. “Data” in this context is the challenge-response pair (CRP) database.

\[
Q = \{ (C_0,R_0), \ldots, (C_k,R_k) \}, \; k < 2^n
\]
Linear Delay Model of Arbiter PUF

[D. Lim, M.S. Thesis, MIT, 2002]

\[
d_{top}(i+1) = \frac{1+C_{i+1}}{2} (p_{i+1} + d_{top}(i)) + \frac{1-C_{i+1}}{2} (s_{i+1} + d_{bottom}(i))
\]

\[
d_{bottom}(i+1) = \frac{1+C_{i+1}}{2} (q_{i+1} + d_{top}(i)) + \frac{1-C_{i+1}}{2} (r_{i+1} + d_{bottom}(i))
\]

where \( C_i \in \{-1,1\} \) denotes the challenge bit of the \( i \)-th stage
Linear Delay Model of Arbiter PUF (contd.)

\[ \Delta(n) = d_{top} - d_{bottom} \]

\[ \Delta(i + 1) = C_{i+1}\Delta(i) + \alpha_{i+1}C_{i+1} + \beta_{i+1} \]

\[ \alpha_n = \frac{p_n - q_n + r_n - s_n}{2} \]

\[ \beta_n = \frac{p_n - q_n - r_n + s_n}{2} \]
Linear Delay Model of Arbiter PUF

Let $p_k$ be the parity of challenge bits:

$$p_i = \prod_{i=k+1}^{n} C_i \quad \text{and} \quad p_n = 1$$

$$\Delta(n) = \alpha_1 p_0 + (\alpha_2 + \beta_1) p_2 + \cdots + (\alpha_n + \beta_{n-1}) p_{n-1} + \beta_n p_n$$

$$= \langle P, D \rangle$$

where $P = (p_0, p_1, \cdots, p_n)$ and $D = (\alpha_1, \alpha_2 + \beta_1, \cdots, \alpha_n + \beta_{n-1}, \beta_n)$

An Arbiter PUF is a linear classifier of random challenge vectors in $n$-dimensional space, where $n$ is the total number of challenge bits.

Apply **Support Vector Machine (SVM)** using:

- Parity vectors $X$ are $n$-dimensional feature vectors
- Constant vector $d$ is the normal to the hyperplane that classifies challenges into two classes.
Modeling Attacks by Machine Learning (Rührmair et al.)

- **Logistic Regression** success rate
  - Arbiter
    - 99.9% using 18K CRPs in 0.6 sec. (64 taps)
  - XOR Arbiter
    - 99% using 12K CRPs in 3 min 42 secs (4 XOR, 64 taps)
  - Lightweight Arbiters
    - 99% using 12K CRPs in 1 hour and 28 mins (4 XORs, 64 taps)
Reported Modeling Attack Results (contd.)

  - Worked on computer simulation model of Arbiter PUF
  - Claimed 100% modeling accuracy by applying SVM (PUF size and training set size not mentioned)

- [Maes *et al*, *IEEE WIFS’12*]
  - Silicon (ASIC) data
  - ASIC fabricated in 65 nm CMOS technology
  - 64-bit Arbiter PUF
  - 500 CRPs as training set
  - Claims ~90% prediction accuracy using SVM

- [CSE Dept., IIT-KGP]
  - Silicon (FPGA Data)
  - 64-bit Arbiter PUF
  - 5000 CRPs as training set
  - ~96% prediction accuracy using SVM
Textbook on Hardware Security

Hardware Security: Design, Threats, and Safeguards

Published: October 29, 2014 by Chapman and Hall/CRC
Content: 542 Pages | 93 B/W Illustrations
Author(s): Debdeep Mukhopadhyay, Rajat Subhra Chakraborty

» Recommend to Librarian (/recommend/librarian/9781439895832)
Thank You for Your Attention!