Cybersecurity Solution in Hardware

Ujjwal Guin

Department of Electrical and Computer Engineering
Auburn University, AL, USA
And this house is even more secure! The front door is four feet thick and made of solid titanium...
Outline

- **Motivation**
  - Counterfeiting and piracy

- **Detection and avoidance of counterfeit ICs**
  - Design-for-Anti-Counterfeit (DfAC) measures
  - Proposed on-chip aging structure
  - End-to-end protection

- **Detection of counterfeit/cloned electronics**
  - Use of Blockchain to enable device traceability
  - Firmware Obfuscation
Counterfeiting and Piracy

Counterfeiting & Piracy
Amount of total counterfeiting globally has reached to 1.2 Trillion USD in 2017 and is bound to reach 1.82 Trillion USD by the year 2020.

Counterfeit Electronics
Electronics companies suffer a loss of around $100 billion every year because of counterfeiting [AGMA].

Counterfeit ICs
Counterfeit components pose a risk of $169B per year for global electronic supply chain [IHS].
Counterfeit Electronic Parts

A counterfeit component [US Dept. Of Commerce, 2010]

- is an unauthorized copy;
- does not conform to original OCM design, model, and/or performance standards;
- is not produced by the OCM or is produced by unauthorized contractors;
- is an off-specification, defective, or used OCM product sold as "new" or working; or
- has incorrect or false markings and/or documentation.
Outline

- Motivation
  - Counterfeiting and piracy

- Detection and avoidance of counterfeit ICs
  - Design-for-Anti-Counterfeit (DfAC) measures
  - Proposed on-chip aging structure
  - End-to-end protection

- Detection of counterfeit/cloned electronics
  - Use of Blockchain to enable device traceability
  - Firmware Obfuscation
Design-for-Anti-Counterfeit (DfAC)

- **Challenges**
  - Process variation impacts detection accuracy.

- **Design goal**
  - Low implementation cost
  - Low cost measurement unit
  - Automated detection process

![Diagram of Reference and Stressed RO sensors](image)

**On-Chip Sensor**

- Loss of detection accuracy

---

**Diagram Description**

- **Reference RO**
  - NAND → INV1 → INV2 → INVn

- **Stressed RO**
  - NAND → INV1 → INV2 → INVn

- **En**
  - Input signal for the ROs

- **$f_R$**
  - Frequency of Reference RO

- **$f_S$**
  - Frequency of Stressed RO

- **$g(\Delta f)$**
  - Distribution of frequency differences

- **$g^{(0)}(\Delta f)$**
  - Baseline distribution of frequency differences

- **$\Delta f$**
  - Frequency difference between $f_R$ and $f_S$
Proposed On-Chip Aging Structure

- Utilize a ring oscillator (RO) and a non-volatile memory (NVM).
- Eliminate the process variation completely.
- NVM Content
  - Frequency ($C_0$), Supply voltage ($V_0$), Temperature ($T_0$), Time duration ($t_{D0}$), Digital signature ($\text{Sig}(H_d)$) on the data $d$. 
Registration Process

Measure RO Frequency ($C_0$), and Read ECID

Construct Data ($D$)

$RD = \{C_0 | T_0 | V_0 | t_{D0}\}$

$D = \{ECID | RD\}$

Program NVM

$\{RD, Sig(H_D)\}$

Compute Hash ($H_D$)

$H_D = Hash_{SHA-2/SHA-3}(D)$

Compute Signature

$Sig(H_D) = E_{Kpri}(H_D)$
Two-Level Authentication Process

- **Level-1**: Hash comparison
  - Applicable for detecting Recycling, Remarking, and cloning.

- **Level-2**: Frequency comparison
  - Applicable for detecting recycling.
Semiconductor Supply Chain

Authenticating a chip at a distributor’s site is extremely challenging

- Distributor may not have the necessary equipment to power up the chip.
- Without powering up a chip prior authentication process is inapplicable.

Potential solution

- Integrate RFID in the chip package.
- Provide authentication using RFID
Proposed End-to-End Protection

- Read RFID Content
  - Extract the information in the tag with a reader.
- Verify RFID Content
  - Perform signature verification.
- Update RFID Content
  - Update its own signature.
Proposed End-to-End Protection - Cont.

Manufacturer (M) → Distributor (D1) → Distributor (D2) → … → Distributor (Dn) → System Integrator (SI)

- Manufacturer (M) provides d, $K_M^+$, and $\text{Sig}_M$ to Distributor (D1).
- Distributor (D1) generates $H_M$, $K_{D1}^+$, and $\text{Sig}_{D1}$.
- Distributor (D1) also receives d, $K_M^+$, and $\text{Sig}_M$ from Manufacturer (M).
- Distributor (D1) sends $H_M$, $K_{D1}^+$, and $\text{Sig}_{D1}$ to Distributor (D2).
- Distributors (D1, D2, …, Dn) follow a similar process.
- System Integrator (SI) receives final $H_D$, $K_S^+$, and $\text{Sig}_D$.
Outline

- Motivation
  - Counterfeiting and piracy
- Detection and avoidance of counterfeit ICs
  - Design-for-Anti-Counterfeit (DfAC) measures
  - Proposed on-chip aging structure
  - End-to-end protection
- Detection of counterfeit/ cloned electronics
  - Use of Blockchain to enable device traceability
  - Firmware Obfuscation
Motivation

- Rapid growth of IoT devices (20-50 billion by 2020)
- Ensuring security and authenticity is a challenge due to severe resource constraints
  - Low power requirements, low area budget, limited memory, and/or extremely low-cost.
  - Traditional cryptographic solutions become infeasible.
- Unsecure supply chain
  - Counterfeit, cloned and tampered devices.
Motivation-Cont.

- Rapid growth of IoT devices (20-50 billion by 2020)
- Ensuring security and authenticity is a challenge due to severe resource constraints
  - Low power requirements, low area budget, limited memory, and/or extremely low-cost.
  - Traditional cryptographic solutions become infeasible.
- Unsecure supply chain
  - Counterfeit, cloned and tampered devices.
Physically Unclonable Functions (PUF)

**SRAM Cell**

During chip fabrication, each MOS transistor experiences process variation and possesses unique electrical properties.

**SRAM Array**

When the SRAM is powered on, each SRAM cell will come up with 1 or 0 randomly across whole array. Some cells will consistently generate 1 or 0 during successive power-ons.

**SRAM Power-up State**

When the SRAM cell is powered up, the SRAM state will be 1 or 0 due to the process variation.

**SRAM PUF ID**

These stable will be the fingerprint (ID) of the device.
Global Identity Blockchain \((B_G)\)

- Aims to provide traceability for edge device
- Across hundreds different manufacturers
- Verify a device without tracking the manufacturer

<table>
<thead>
<tr>
<th>Public ID</th>
<th>Private ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>HID_{A1}</td>
</tr>
<tr>
<td>B1</td>
<td>HID_{B1}</td>
</tr>
<tr>
<td>A2</td>
<td>HID_{A2}</td>
</tr>
<tr>
<td>C1</td>
<td>HID_{C1}</td>
</tr>
<tr>
<td>N1</td>
<td>HID_{N1}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>B2</td>
<td>HID_{B2}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Ledger

Authenticator

B2

Yes

Authentic

HID^{*}_{B2}

No

Non-Authentic

Market

B2

Authenticator

HID_{B2}

\(B_G\)

Manufacturer of Device A

Manufacturer of Device B

Manufacturer of Device C

Manufacturer of Device N
Local Identity Blockchain ($B_L$)

- Locally *permissioned* blockchain infrastructure
  - A local administrator can register an authentic edge device.
- IoT Devices can be authenticated periodically to prevent device cloning
  - A device could be compromised during lifespan
- A secret key is generated and burned into the on-chip, one-time programmable memory.
  - Once the key is programmed, direct access of device ID is prohibited to protect it from being copied.
Local Identity Blockchain

Market

Local Admin

Program Key

Generate Key

Registration

New Device

TLS

A2

\(K_{A2}\)

\(EK_{A2}\)

\{A2, HID_{A2}, EK_{A2}\}

G1

G2

G3

SCP

D3

D1

D2

D3

HID_{D3}

EK_{D3}

BL

Ledger

<table>
<thead>
<tr>
<th>Public ID</th>
<th>Hash of Private ID</th>
<th>Encrypted Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>A2</td>
<td>HID_{A2}</td>
<td>EK_{A2}</td>
</tr>
<tr>
<td>B1</td>
<td>HID_{B1}</td>
<td>EK_{B1}</td>
</tr>
<tr>
<td>D3</td>
<td>HID_{D3}</td>
<td>EK_{D3}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>B2</td>
<td>HID_{B2}</td>
<td>EK_{B2}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
SCP: Secure Communication Protocol

- Securely transfer PUF ID to the gateway.
- Lightweight and can be implemented in the edge devices.

**Gateway**

1) Generate $n$, $n \in \{0,1\}^N$
2) Compute $m_i$, $m_i = K_i \oplus n$
3) Send $m_i$

10) Receive $r_i$
11) Compute $H$, $H = \text{hash}(n)$
12) Reconstruct secret device ID
   
   $$ID_r = r_i \oplus H$$
   $$= ID_i \oplus H \oplus H$$
   $$= ID_i$$
13) Compare the hashes of reconstructed ID ($HID_r$) and stored ID ($HID_i$)

**Edge Device ($ED_i$)**

4) Receive $m_i$
5) Recover $n$, $m_i \oplus K_i$
   $$= K_i \oplus n \oplus K_i = n$$
6) Compute $H$, $H = \text{hash}(n)$
7) $ID_i \leftarrow \text{PUF Response}$
8) Compute $r_i$
   $$r_i = H \oplus ID_i$$
9) Send $r_i$
Firmware Obfuscation

- Microcontroller STM32F4 has two levels of memory protection to defend against firmware extraction.
- Firmware integrity checking (e.g. CRC) during start-up can be used to bypass the level-I security.
- Even level-II security can be bypassed using UVC.
- Firmware can be copied, which makes cloning easy.

Experimental Setup to extract the firmware from STM32F4
Firmware obfuscation process

1. Read Device ID
2. Create Key
3. Program Key to a secure NVM
4. Load Obfuscated Firmware
Firmware Reconstruction Process

Instruction

Power-up

Bootloader

$K_0$

PUF

$K$

Non-volatile Memory (NVM)

Address

Address

0

1

S

Processor Core

Tag

Data

$A_{01}[31:7]$   $I_{O2}$

$A_{02}[31:7]$   $I_{O1}$

...  ...

$A_{02L}[31:7]$   $I_{O2L-1}$

Instruction Address [6:2]

Non-volatile Memory (NVM)
Conclusion

- Counterfeiting and piracy
- Design-for-Anti-Counterfeit (DfAC) measures for detecting counterfeit ICs
  - On-chip aging structure
  - End-to-end protection for component supply chain
- Device authentication scheme using Blockchain technology to detect clones
  - Global Blockchain Infrastructure ($B_G$) for implementing Traceability of IoT edge devices
  - Local Blockchain ($B_L$) for edge device registration and authentication
  - SCP: Secure Communication Protocol for edge device authentication
- Firmware obfuscation to prevent cloning
Thank you!

Any Questions?