Automatic Extraction of Secrets from the Transistor Jungle using Laser-Assisted Side-Channel Attacks

Thilo Krachenfels*, Tuba Kiyan*, Shahin Tajik† and Jean-Pierre Seifert*

* Technische Universität Berlin, Chair of Security in Telecommunications, Germany
† Worcester Polytechnic Institute, Department of Electrical and Computer Engineering, USA

30th USENIX Security Symposium
August 11–13, 2021
Overview

• Root-of-trust keys stored in tamper/read-proof non-volatile memories
• While in use: key is loaded into on-chip registers/memories
• These memory cells can be read out using sophisticated optical side-channel analysis techniques
Overview

• Chip manufacturers believe that the design complexity and high reverse-engineering costs are a natural barrier against attackers

• **Goal of this work:** Show that secret extraction from CMOS memory cells is possible...
  • ... without the need to reverse-engineer the chip’s layout and memory mapping
  • ... in an **automated** fashion
Outline

1. Overview
2. Background
3. Approach
4. Experimental Setup
5. Results
6. Conclusion
Background
Logic state extraction through the chip backside

• Using techniques known from IC failure analysis

• Silicon backside is transparent to near-infrared (NIR) light

• Light can enter the silicon and
  • ... generate localized heat (→ Thermal laser stimulation)
  • ... is influenced by electric properties and reflected (→ Optical probing)
Background

Thermal Laser Stimulation (TLS)

• Local heat generation by laser irradiation
• Temperature gradient creates voltage ("Seebeck voltage")
• Changed voltage causes data-dependent current flow which can be measured
→ Grayscale-encoded “map” of the chip
Background
Laser Logic State Imaging (LLSI)

• Optical probing technique through the chip backside
• Modulate supply voltage → reflected light is modulated
• On- and off-state transistors can be distinguished
• Can extract logic states of memory cells → Grayscale-encoded image of transistor states

CMOS memory cell

LLSI signal origin
Power supply modulation
Low/High-ohmic channel

VCC
M_{p1}
M_{n1}
M_{p2}
M_{n2}
GND
Approach

(1) Automated Measurements

(2) Neural Network Training

(3) Secret Extraction

\[ S_x = [b_1, b_2, \ldots, b_n] \]
Setup

Optical and electrical setup

- Hamamatsu PHEMOS-1000 failure analysis microscope
  - Different lenses (5x, 20x, 50x)
  - Light source for optical probing
  - Laser for thermal stimulation
  - Automation capabilities
Setup

Learning setup

• Use convolutional neural networks (CNNs)
  • ... to prove that the secret can be extracted without any manual tuning and knowledge about the layout of the chip
  • ... with a very simple model and data augmentation techniques

• Other (classical) machine learning techniques should be applicable as well
Setup

Devices under test

• Targets for secret extraction

Dedicated key memory (BBRAM) of a *Xilinx UltraScale FPGA*

Key in flip-flops of an *Intel Cyclone IV FPGA*

Key in SRAM of a *Texas Instruments MSP430* microcontroller
Image Dataset

• Automatic capturing of LLSI and TLS images

<table>
<thead>
<tr>
<th>Target</th>
<th># Mem. bits</th>
<th># Key bits</th>
<th>Technique</th>
<th>Image dimensions</th>
<th>Lens and scanner zoom</th>
<th># Images</th>
<th>Time/Image (mm:ss)</th>
<th>Total time (hh:mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBRAM</td>
<td>288</td>
<td>256</td>
<td>TLS</td>
<td>985 px × 407 px</td>
<td>50× (×2)</td>
<td>578</td>
<td>02:02</td>
<td>19:35</td>
</tr>
<tr>
<td>MSP430 (zeroized)</td>
<td>8192</td>
<td>512</td>
<td>LLSI</td>
<td>503 px × 355 px</td>
<td>50×</td>
<td>433</td>
<td>13:00</td>
<td>93:49</td>
</tr>
<tr>
<td>MSP430 (randomized)</td>
<td>8192</td>
<td>512</td>
<td>LLSI</td>
<td>503 px × 355 px</td>
<td>50×</td>
<td>821</td>
<td>13:00</td>
<td>177:53</td>
</tr>
<tr>
<td>FPGA Registers</td>
<td>16</td>
<td>16</td>
<td>LLSI</td>
<td>509 px × 28 px</td>
<td>50× (×2)</td>
<td>568</td>
<td>2:40</td>
<td>25:17</td>
</tr>
</tbody>
</table>

Dataset available online:
http://dx.doi.org/10.14279/depositonce-11354
Image capturing

Xilinx Ultrascale: 256-bit key

MSP430: 512-bit key in 8192-bit SRAM

Rest zeroized

Rest randomized
Results

• First experiments: trying to learn a single key bit from the full image
  → Successful
Results

• Learning multiple bits in parallel
  • One output neuron per bit
  → Successful if applied on smaller fractions of the image
Results

- Number of required images
  - For smaller image fractions, less training data is necessary
Results

• Localization of bits

![Localization of bits diagram](image-url)
Results

- Successful key extraction also from the other two devices
- More details: please take a look at our paper.

<table>
<thead>
<tr>
<th>Section</th>
<th>Bit positions (most significant bit first)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>7, 15, 23, 31, 39, 47, 55, 63, 71, 79, 87, 95, 103, 111, 119, 127, 135, 143, 151, 159, 167, 175</td>
</tr>
<tr>
<td>8</td>
<td>374, 382, 390, 398, 406, 414, 422, 430, 438, 446, 454, 462, 470, 478, 486, 494, 502, 510</td>
</tr>
<tr>
<td>10</td>
<td>6, 14, 22, 30, 38, 46, 54, 62, 70, 78, 86, 94, 102, 110, 118, 126, 134, 142, 150, 158, 166, 174, 190</td>
</tr>
<tr>
<td>14</td>
<td>373, 381, 389, 397, 405, 413, 421, 429, 437, 445, 453, 461, 469, 477, 485, 493, 501, 509</td>
</tr>
</tbody>
</table>
Conclusion

• Sophisticated HW attacks often seen as too costly and time-consuming to pose a severe threat

• We have shown: Automation of FA tools combined with machine learning makes tedious reverse-engineering superfluous

→ Countermeasures required when designing new devices for critical applications

Preprint
https://arxiv.org/abs/2102.11656
Thanks for your attention!

Contact for questions: tkrachenfels [AT] sect.tu-berlin [DOT] de