Glitching and Side-Channel Analysis for All

Colin O’Flynn – NewAE Technology Inc.

RECON 2015 – Montreal, QC.
Overview

• W.t.f is side-channel power analysis (again)
• Example: IEEE 802.15.4 Node
• Example: AES-256 Bootloader
• W.t.f. is Glitching
• Simple power glitching
About Me

• PhD at Dalhousie University in Halifax, Canada (Ongoing)
• Designed open-source hardware security project (ChipWhisperer)
• Commercialization through NewAE Technology Inc.
• Previously talked at Blackhat US/EU/AD, RECON, ESC
Side Channel Power Analysis
Side Channel Analysis

Plaintext → Crypto Device → Ciphertext

Secret Key
Super-Fast Side Channel
Real-Life

Average Measurement vs. Hamming Weight of Leakage
Breaking Apart
Hardware Example
Hackaday Prize 2014

ChipWhisperer®: Security Research

ChipWhisperer laughs at your AES-256 implementation. But it laughs with you, not at you.

cof lyn

DESCRIPTION

ChipWhisperer is the first open-source toolchain for embedded hardware security research including side-channel power analysis and glitching. The innovative synchronous capture technology is unmatched by other tools, even from commercial vendors. Similar commercial equipment is too expensive ($30k+), and being closed-source limits usefulness for academics. Instead this project bridges the gap between academic research and in-the-trenches engineering. Several peer-reviewed publications describe the design, matched with hours of hands-on tutorials for getting started.

The objective of ChipWhisperer is nothing short of revolutionizing the entire embedded security industry. Every designer who uses encryption in their design should be able to perform a thorough security analysis on it. Today, this is extremely expensive and out of reach for many due to the limitations of commercial hardware. ChipWhisperer opens it up to a much larger audience.
Cheap Hardware... First Ver

ChipWhisperer™

The first open-source hardware security analysis tool.
ChipWhisperer-Lite Kickstarter

ChipWhisperer-Lite: A New Era of Hardware Security Research

Embedded security - is it an oxymoron? Learn the truth through a series of hands-on labs targeting computer and electrical engineers.

Created by
Colin O’Flynn

331 backers pledged $88,535 to help bring this project to life.
Cheaper Hardware

Atmel SAM3U
High-Speed USB

Micro-USB (Power + Comms)

Xilinx S6LX9 FPGA
Break-away connections

10-bit ADC (105 MS/s)
+55dB Low Noise Amplifier

XMEGA Target
Open-Source Software

Certain files in this project may have specific licenses or copyright restrictions, as this project uses multiple open-source projects. Files in this project without any specific license can be assumed to follow the following general clause:

ChipWhisperer is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.
Example of Power Analysis

<demo here>
IEEE 802.15.4 Nodes
IEEE 802.15.4
Example #1: 802.15.4

ZigBee (ZigBee IP, ZigBee Pro, RF4CE, etc.)
WirelessHART
MiWi
ISA100.11a
6LoWPAN
Nest Weave
JenNet
Thread
Atmel Lightweight Mesh
IEEE 802.15.5
DigiMesh

http://eprint.iacr.org/2015/529
Hardware Setup
802.15.4 Frame Format

Frame Header

Seq. Number

Dest Address ([ff = Broadcast])

Source Addressing

Sec. Level.

FrameCounter

Key ID

Encrypted Payload + MAC (MIC in 802.15.4 parlance)

CRC-16 Goes Here
IEEE 802.15.4 Wireless Stack: Frame Decryption Procedure:

1. Validate headers and security options.
2. Check that the received frame counter is numerically greater than the last stored frame count.
3. Look up the secret key based on message address and/or key index.
4. Decrypt the payload (and MAC if present).
5. Validate the MAC (if present).
6. Store the frame counter.
Example #1: 802.15.4

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flags</td>
<td>Source Long Addr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Addr (cont’d)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Addr (cont’d)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F.C.</td>
<td>SecLevel</td>
<td>AES Counter</td>
<td></td>
</tr>
<tr>
<td>(cont’d)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Many fixed bytes...
CPA Attack Result

\[ p^1 = [c\ c\ c\ c\ c\ c\ c\ c\ X\ X\ X\ X\ c\ c\ c]\]

\[ r^1 = [c\ c\ c\ c\ c\ c\ c\ c\ c\ c\ K*K*K*K*\ c\ c\ c]\]

\[ v^1 = [c\ c\ Y*\ c\ c\ Y*\ c\ c\ c\ c\ c\ c\ Y*\ c\ c\ Y*]\]

\[ m^1 = [Z*Z*Z*Z*Z*Z*Z*Z*\ c\ c\ c\ c\ Z*Z*Z*Z*]\]
Example #2: AES-256 Bootloader

Tutorial:
http://newae.com/sidechannel/cwdocs/tutorialaes256boot.html

Paper (CCECE 2015):
Bootloader Protocol
AES-256 in CBC Mode

![Diagram showing AES-256 decryption in CBC mode](image-url)
Round 14
Round 13
Trace View

Synchronization of Power Traces

Sample Number

Power Trace

Sample Number
Success Rate

AES-256 Attack Success

- **Global Success Rate**
- **Number of Traces**

- **Round 14 Key**
- **Round 13 Key**
Getting Started in Side Channel Power

• Build/buy a simple target device:
  • AVR dev-board
  • Arduino Uno
  • PIC

• Get a scope with USB API
  • Picoscope
  • Most bench scopes
  • Be wary of cheap off-brand scopes, sometimes USB interface is poor

• Experiment!
Glitching
Glitching Target

```c
int i, j, count;

while(1){
    count = 0;

    for (j = 0; j < 5000; j++){
        for (i = 0; i < 5000; i++){
            count++;
        }
    }
}

printf("%d %d %d\n", count, i, j);
```
Easy Glitching
High-Precision Glitches
Easy Glitching
Raspberry Pi Example
Raspberry Pi Example
Glitch Tool
Glitch Waveform (Raspberry Pi)
Getting Started in Glitching

• Load simple code onto target
• Determine/guess sensitive power rail
• Test glitch parameters → ideally with profiling code
Glitching in CW-Lite
It’s fun!

Try Power Analysis and Glitching today!

ChipWhisperer Project: www.chipwhisperer.com

NewAE Technology Inc.: www.newae.com

Personal:

@colinoflynn
coflynn@newae.com
http://www.oflynn.com