A Hardware Approach to Information Flow Security

Jason Oberg and Ryan Kastner

{jkoberg, kastner}@cs.ucsd.edu

Problem: Hardware Leaks Information

Applications
Prog. Language
Compiler/OS
Instruction Set
Microarchitecture
Functional Units
Logic Gates

Side and Covert Channels

Cache Power/Side Channel

CPU Cache
AES

Timing
Secret
Power

- Varying latency based on key
- Power differences based on key

Bus Covert Channel

Arbiter

Timing

- Bus Contention
- Internal State can leak information
- Eliminating covert channels not obvious

Solution: Tools for Hardware Designers

Gate Level Information Flow Tracking (GLIFT)
Bit-level tracking
Detects timing channels

Overview

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>GLIFT</th>
<th>Other</th>
<th>Covert-Seq</th>
<th>Static-Sim</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>482.0</td>
<td>17619.0</td>
<td>13619.0</td>
<td>13619.0</td>
</tr>
<tr>
<td>CPU Cache</td>
<td>131.1</td>
<td>254013.5</td>
<td>254013.5</td>
<td>254013.5</td>
</tr>
<tr>
<td>C515</td>
<td>240128.5</td>
<td>-</td>
<td>-</td>
<td>91069.0</td>
</tr>
<tr>
<td>C992</td>
<td>240128.5</td>
<td>-</td>
<td>-</td>
<td>91069.0</td>
</tr>
<tr>
<td>USB</td>
<td>155325.9</td>
<td>-</td>
<td>-</td>
<td>1140575.9</td>
</tr>
<tr>
<td>I2C</td>
<td>155325.9</td>
<td>-</td>
<td>-</td>
<td>1140575.9</td>
</tr>
<tr>
<td>Core</td>
<td>155325.9</td>
<td>-</td>
<td>-</td>
<td>1140575.9</td>
</tr>
</tbody>
</table>

Testing Method

- Prototype Secure System
  - Use GLIFT to prove strong non-interference between devices in I2C and USB
  - Varying trust devices to all exist on a shared bus without any digital interference.

- Information Flows in I2C and USB
  - GLIFT used to show non-interference between devices in I2C and USB
  - Varying trust devices to all exist on a shared bus without any digital interference.

Current Work

- Formal Model for how GLIFT tracks timing channels
- Differential Power Analysis (DPA)
- Better HDLs for building secure hardware.

