Power Analysis Part II.a

Cryptographic Hardware for Embedded Systems

ECE 3894

Fall 2019

Assoc. Prof. Vincent John Mooney III

Georgia Institute of Technology

©Georgia Insitute of Technology, 2018-2019
Reading


- All figures in this lecture are from the aforementioned manuscript.
Questions Answered by This Lecture

• If one measures the energy consumption or power of a microchip, what do the power traces reveal?
• What is the statistical methodology used to reveal the information claimed to have been learned?
Power Analysis

- Cryptographic operations executed on microchips exhibit variations in energy consumption / power
  - Note that power is a rate of energy consumption, i.e., joules per second
- The energy consumption of microchip implementations of cryptography depends on the data values ($P_{data}$) and the specific operations (typically mathematical or memory storage) performed ($P_{op}$)
- There is also an electrical noise component ($P_{el.\, noise}$) and constant ($P_{const}$)
- $P_{total} = P_{op} + P_{data} + P_{el.\, noise} + P_{const}$
- This model may be refined for certain situations, but suffices for most
- Note that cryptographic information may be revealed by $P_{op}$ and $P_{data}$
Figure 3.9. Picture of the measurement setup for the attacks on the 8-bit microcontroller.
Example Power Measurement Setup

• The previous slide shows the microcontroller power measurement setup of Mangard et al.
• Microcontroller power supply is 5V and clock frequency 11 MHz
• Voltage drop across a 1 Ω resistor connected to the power supply (Gnd) is measured by an oscilloscope
• Oscilloscope can sample 8 bits of resolution every nanosecond (GHz)
• Typically measure every four ns in experiments (250 Million Samples per second or MS/s)
Consider a Single Point in a Power Trace

• A moment in time
• Aim to determine the probability distributions of $P_{op}$, $P_{data}$ and $P_{el.\,noise}$
\textbf{Figure 4.1.} Power traces look very similar if the same data is processed. \textbf{Figure 4.2.} Histogram of the power consumption at 362 ns of Figure 4.1.
Figures 4.1 and 4.2 from Mangard et al.

- Fig. 4.1 shows five power traces with the same data and instruction (a load of a zero from on-chip memory to a register)
- The differences in the power traces are due to noise
- Fig. 4.2 shows a histogram of 10,000 power traces of the same operation considering the Voltage across the resistor (recall P=IV and I is typically a constant)
  - Most of the measurements are near 112 mV
  - Very few measurements are below 109 mV or above 115 mV
Gaussian (a.k.a. Normal) Distribution

- $f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2} \left(\frac{x-\mu}{\sigma}\right)^2}$
- $\mu = E(X)$
- $\sigma^2 = Var(X) = E((X - E(X))^2)$
- $X \sim N(\mu, \sigma)$ means that $X$ is normally distributed (has a Gaussian distribution) with mean value $\mu$ and standard deviation $\sigma$
- The “standard normal distribution” has $\mu = 0$ and $\sigma = 1$
Back to Our Experiment at 362 ns

- We can estimate $\mu = E(X)$ with the average $\bar{x}$ calculated empirically:
  - $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$

- We can estimate $\sigma = \sqrt{Var(X)}$ with the square root of the variance also calculated empirically:
  - $s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2$

- For this experiment, the following is calculated from the 10,000 traces
  - $\mu = \bar{x} = 111.86 \text{ mV}$
  - $\sigma = s = 1.63 \text{ mV}$

- Thus, $X \sim N(111.86, 1.63)$ where the units are millivolts
Figure 4.3. The normal distribution $\mathcal{N}(111.86, 1.63)$ models the power consumption at 362 ns.
Power Analysis So Far

- From our current experiment consisting of 10,000 execution of the same memory operation with a data value of zero, we find:
  - $E(P_{op}) = E(P_{data}) = E(P_{const}) = 0$
  - $P_{el.\ noise} \sim N(0, 1.63)$
Next: Data Dependence

• Instead of moving a constant zero from memory to a register location as was done for estimating the distribution of noise on the power pin of this microcontroller, we now vary the eight bit memory data value among all 256 possible values.

• 200 measurements for each value = 256*200 = 51,200 total measurements.

• Again, let us look at what happens at 362 ns.

• (Quick note before we continue: this discussion will assume large amounts of encrypted data values which are evenly distributed among all possibilities, e.g., zero does not occur more often than any other number; another way to say this is that data values are uniformly distributed.)
Figure 4.4. Histogram of the power consumption at 362 ns if different data values are transferred from the internal memory to a register.
Histogram Comments on Figure 4.4

• Not describable with a single Gaussian
• However, upon further inspection, there do appear to be nine Gaussians
• Why could this be?
  • Answer: consider the Hamming Weight (HW) between each possible data value and zero
HW Calculations on 8 bits Values

- HW 0: 00000000 (TOTAL: 1)
- HW 1: 00000001, 00000010, …, 10000000 (TOTAL: 8)
- HW 2: 00000011, 00000101, …, 10000001, 100000010, …
- …
- HW 7: 11111110, 11111101, …, 01111111 (Total: 8)
- HW 8: 11111111 (TOTAL 1)
- Result is a binomial distribution
  - Probably of HW 4 is 27.3%, HW 3 or 5 have an equal probability of 21.9%, …, HW 1 or 7 have a probability of 0.031%, HW 0 or 8 have probability 0.004%
HW Calculations on 8 bits Values (Distance from 11111111)

- HW 8: 00000000 (TOTAL: 1)
- HW 7: 00000001, 00000010, ..., 10000000 (TOTAL: 8)
- HW 6: 00000011, 00000101, ..., 10000001, 10000000, ...
- ...
- HW 1: 11111110, 11111101, ..., 01111111 (Total: 8)
- HW 0: 11111111 (TOTAL 1)

- Result is a binomial distribution
  - Probably of HW 4 is 27.3%, HW 3 or 5 have an equal probability of 21.9%, ..., HW 1 or 7 have a probability of 0.031%, HW 0 or 8 have probability 0.004%
Explanation for Figure 4.4

- Function of HW
  - The greater the HW, the greater the energy consumption \( P_{\text{data}} \)
  - Add in \( P_{\text{el. noise}} \)

- Load and store instructions in any ISA are known to have a large HW energy consumption component due to the way SRAMs are designed
  - Rows and columns have to be charged and discharged in an array of 6T single bit memory elements
  - Sense amplifiers are used to detect when a 1 is overpowered by a 0 and vice-versa
  - Furthermore, for this microcontroller, bus lines are precharged to 1 each time

©Georgia Institute of Technology, 2018-2019
Question: How to Remove the Noise?

• Calculate the mean voltage for each HW
• In this example, find 111.9, 117.6, 123.2, 128.7, 134.0, 139.5, 145.1, 151.2 and 159.6 (all in mV)
• As was done for 111.9 (i.e., HW 0) where we found $\sigma = s = 1.63 \text{ mV}$, we can similarly find the standard deviation for each HW
Result

• Can superpose nine Gaussian distributions to accurately model the measurements
*Figure 4.5.* The distribution of the power consumption when the microcontroller transfers different data from the internal memory to a register.
Next: Energy Consumption per Operation

• Similar to previous, but now alter the ISA operation type
  • Some operations, e.g., load and store or add and subtract, can be grouped together
  • Some operations are very specific, e.g., floating point multiply
  • Also may have to account for multicycle operations
  • As stated in the book, the result is that $P_{op}$ can also be approximated reasonably accurately for this microcontroller (and many other instruction-set architectures or ISAs) by a Gaussian distribution
Signal to Noise Ratio (SNR)

• Two questions: what information is the attacker seeking and what do the points of a power trace provide towards revealing this information?

• We begin with a distinction between exploitable power measurement data $P_{\text{exp}}$ which must be due to either $P_{\text{data}}$ or $P_{\text{op}}$
  
  • The aspects of $P_{\text{data}}$ and/or $P_{\text{op}}$ which may not be exploitable, e.g., due to various bits switching back and forth not under observation are called $P_{\text{switching}}$ or $P_{\text{sw. noise}}$

• Therefore, $P_{\text{exp}} + P_{\text{switching}} = P_{\text{data}} + P_{\text{op}}$

• Furthermore, $P_{\text{total}} = P_{\text{exp}} + P_{\text{switching}} + P_{\text{el. noise}} + P_{\text{const}}$
SNR Continued

\[ SNR = \frac{\text{Var}(\text{Signal})}{\text{Var}(\text{Noise})} \]

\[ SNR = \frac{\text{Var}(P_{\text{exp}})}{\text{Var}(P_{\text{switching}} + P_{\text{el. noise}})} \]
Example

• A processor operates on an 8-bit value where each bit is independent and uniformly distributed

• Assume that the value of the second bit is always the complement of the first bit in the experiments carried out
  • E.g., 0bX₁X₂X₃X₄X₅X₆X₇ and 0bY₁Y₂Y₃Y₄Y₅Y₆Y₇ where the first bit considered in our analysis is the case of the LSB = 0 and the second bit considered in our analysis is the case of the LSB = 1
  • The other 14 bits are independent and uniformly distributed

• $P_{\text{exp}}$ consists of the energy consumed by the LSB

• $P_{\text{switching}}$ consists of the energy consumed by the rest of the bits
Example (continued)

• We have 51,200 power traces as computed already earlier
• Select the 25,600 traces with LSB = 1
• Figure 4.6 shows the resulting histogram at 362 ns
Figure 4.6. Histogram of the total noise ($P_{sw.\ noise} + P_{el.\ noise}$) if the exploitable signal is the LSB of the byte that the microcontroller processes. This noise is approximately normally distributed.
Comments

- $P_{\text{switching}}$ in Fig. 4.6 has a binomial distribution
- Fig. 4.6 also includes $P_{\text{el. noise}}$
- However, we can approximate Fig. 4.6 with a single Gaussian as shown
  - The result is $\sigma = s = 7.54 \text{ mV}$; thus, for the LSB (1-bit) scenario, we find that $\sigma$ of $P_{\text{switching}} + P_{\text{el. noise}} = 7.54 \text{ mV}$
  - Hence, $\text{Var}(P_{\text{switching}} + P_{\text{el. noise}}) = (7.54 \text{ mV})^2 = 56.85 \text{ mV}^2$
  - Also, earlier we found that $\sigma$ of $P_{\text{el. noise}}$ for one bit = $1.63 \text{ mV}$
  - Hence, $\text{Var}(P_{\text{el. noise}}) = (1.63 \text{ mV})^2 = 2.67 \text{ mV}^2$
Table 4.2. Variance of the components of the power consumption according to the models discussed in (4.1) and (4.8).

<table>
<thead>
<tr>
<th>Component</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>8-bit scenario</td>
</tr>
<tr>
<td>$P_{data}$</td>
<td>61.12</td>
</tr>
<tr>
<td>$P_{op}$</td>
<td>0.00</td>
</tr>
<tr>
<td>$P_{el.,noise}$</td>
<td>2.67</td>
</tr>
<tr>
<td>$P_{exp}$</td>
<td>61.12</td>
</tr>
<tr>
<td>$P_{sw.,noise} + P_{el.,noise}$</td>
<td>2.67</td>
</tr>
</tbody>
</table>
Comparison

• SNR is much higher for 8-bits than for 1-bit
Figure 4.7. The signal levels, the standard deviation of the noise, and the SNR when attacking a uniformly distributed 8-bit data value on our microcontroller.
Figure 4.7. The signal levels, the standard deviation of the noise, and the SNR when attacking a uniformly distributed 8-bit data value on our microcontroller.