The basic concepts of power analysis attacks are reviewed. Various countermeasures against these attacks are presented and their weaknesses are discussed. One promising software countermeasure that uses random masks is more thoroughly investigated. A second-order attack against this countermeasure is introduced and an optimal decision threshold is discussed.
Power Analysis Attack Countermeasures and Their Weaknesses

Presented By:

Thomas S. Messerges

Security Technology Research Laboratory
Motorola

Tom.Messerges@motorola.com

October 12, 2000
CEPS – Communications, Electromagnetics, Propagation, & Signal Processing Workshop
University of Illinois at Urbana-Champaign
Summary of Presentation

• Review of Basic Concepts
  • Smartcards
  • Example attack

• Hardware Countermeasures
  • Noise generator
  • Power signal filtering
  • Novel circuit designs

• Software Countermeasures
  • Time randomization
  • Masking techniques

• Attacking Masking Countermeasures
Smartcard Overview

- A smartcard is:
  - a plastic card with an embedded microprocessor
  - “secure” against malicious tampering and monitoring

- Typical smartcard processor:
  - 8-bit CPU, 384 bytes RAM, 24K ROM, 8K EEPROM, 3 to 5 Mhz clock rate

- Newest smartcards:
  - 32-bit RISC, 4 Kbytes RAM, 96K ROM, 64K EEPROM, 50 Mhz clock rate

Pin Numbering:

1  5
2  6
3  7
4  8

Bob Smith

Smartcard
Power Analysis Attacks (Kocher et al. - Jun. 1998)

- Measure instantaneous power consumption of a device while it runs a cryptographic algorithm:

Different power consumption when operating on logical ones compared to operating on logical zeros
Example of Power Consumption Information Leakage

Hamming Weight or Hamming Distance Leakage

Voltage

Time
Example of a Vulnerable Algorithm

Vulnerable to first-order DPA attack

Example attack on the Twofish whitening process:

S. Chari, C. Jutla, J.R. Rao, and P. Rohatgi:
“A Cautionary Note Regarding Evaluation of AES Candidates on Smart-Cards,”
Sort the Signals to Extract 1st-Order Biases

\[ P_i = 1 \quad P_i = 0 \]

Set \( S_1 \)

Set \( S_0 \)

Sorting the signals into two sets may introduce biases

Motorola Labs
Sort the Signals to Extract 1st-Order Biases

\[ P_i = 1 \quad P_i = 0 \]

Set \( S_1 \)  

Set \( S_0 \)

Peaks are: \textit{HIGH} when \( \text{Result}_i = K_i \oplus P_i = 1 \) hence \( K_i = 0 \)
Sort the Signals to Extract 1st-Order Biases

\[ P_i = 1 \]

Set \( S_1 \)

\[ P_i = 0 \]

Set \( S_0 \)

Peaks are: LOW

when (Result\(_i = K_i \oplus P_i = 0\) hence (K\(_i = 0\))
Sort the Signals to Extract 1st-Order Biases

\[ P_i = 1 \quad \text{Set } S_1 \]

\[ P_i = 0 \quad \text{Set } S_0 \]

Need to determine which set has **HIGH** peaks and which has **LOW** peaks in the presence of noise.
Average and Take the Difference to Expose Biases

$$A_1(\text{Average Signal from } S_1) :$$

$$-$$

$$A_0(\text{Average Signal from } S_0) :$$

$$=\text{DPA Bias Signal}$$

$$T(n) = A_1(n) - A_0(n)$$

Spike confirms a correct key guess
Noise Generator

• **Power Randomization** (Daemen and Rijmen ‘99)

![Noise Waveform]

• **Advantage:**
  - Design may be relatively simple
  - Effective way to resist attacks

• **Disadvantages:**
  - Expensive to implement
  - Not always possible for legacy systems
  - Might be easy to disable through tampering
  - Not energy efficient
  - Signal is still present
Power Signal Filtering

• **Active Power Filter**  (Rakers et al, ‘00)

- **Advantage:**
  - Design may be relatively simple
  - Effective way to resist attacks

- **Disadvantages:**
  - Requires a change to the hardware
  - Might be easy to disable through tampering
  - **Passive filter:** Physical limitations restrict the size of an on-chip capacitor (Shamir ‘00 and Coron et al. ‘00)
  - **Active filter:** Compensation techniques are likely to lag behind power supply changes (Shamir ‘00 and Coron et al. ‘00)
Novel Circuit Designs

- **Detachable Power Supplies** (Shamir, ‘00)

![Circuit Diagram]

- **Advantage:**
  - Design may be relatively simple
  - Effective way to resist attacks

- **Disadvantages:**
  - Not always practical for legacy systems
  - Susceptible to active attacks
  - Signal can leak via other means
Time Randomization

- **Desynchronization** (Daemen and Rijmen ‘99)

  ![Diagram of desynchronization]

  - **Advantages:**
    - Easy and cheap to implement
    - Increases difficulty of attack

  ```
  if (random bit equals 1)
  NOP;
  
  r = random bit (either 0 or 1);
  result[r ⊕ 0] = data[r ⊕ 0];
  result[r ⊕ 1] = data[r ⊕ 1];
  ```

  - **Disadvantage:**
    - Susceptible to signal processing analysis
Masking Techniques

- **Duplication** (Goubin and Patarin ‘99, Messerges ‘00)

  00111101 10110110
  10001011

  2 - Shares

- **Advantage:**
  - Eliminates the threat of 1st-order DPA
  - Attackers need to mount 2nd-order DPA attacks

- **Disadvantage:**
  - Some cryptographic functions are difficult to mask
  - Susceptible to 2nd-order DPA
Example of a Masking Countermeasure

\[ w_1(PTI) \]
\[
\{ \\
A: \text{Result} = PTI \oplus \text{SecretKey} \\
\ldots \\
\text{more operations} \ldots \\
\ldots \\
\text{return } CTO
\}
\]

\[ w_2(PTI) \]
\[
\{ \\
B: \text{RandomMask} = \text{rand()} \\
mPTI = PTI \oplus \text{RandomMask} \\
C: \text{Result} = mPTI \oplus \text{SecretKey} \\
\ldots \\
\text{more masked operations} \ldots \\
\ldots \\
\text{unmask and return } CTO
\}
\]

Vulnerable to second-order DPA attack where 2 samples are examined
Sort the Signals to Extract 2nd-Order Biases

Set \( S_1 \)

Set \( S_0 \)

Sorting the signals into two sets may introduce biases
Sort the Signals to Extract 2nd-Order Biases

Set $S_1$

Peaks are: **HIGH** and **LOW**

Set $S_0$

Peaks are: **HIGH** and **LOW**
Sort the Signals to Extract 2nd-Order Biases

Set $S_1$

Set $S_0$

Peaks are: HIGH and LOW

Peaks are: HIGH and LOW
Sort the Signals to Extract 2nd-Order Biases

Set $S_1$

Set $S_0$

Peaks are: CORRELATED
Sort the Signals to Extract 2nd-Order Biases

Peaks are: INVERSELY CORRELATED
Sort the Signals to Extract 2nd-Order Biases

Set $S_1$

Peaks are: **CORRELATED**

Set $S_0$

Peaks are: **INVERSELY CORRELATED**

Need to determine which set is correlated and which set is uncorrelated in the presence of noise.
How to Distinguish “Inversely Correlated” from “Correlated”

• Ad hoc statistical approach:

Calculate statistic: \[ S = \sum_{k=0}^{N-1} |b_k - c_k| \]

\( S \) should be smaller when \( b_k \) and \( c_k \) are correlated.

• Optimal statistical approach:

• Gaussian and independence assumption for \( b_k \) and \( c_k \) leads to an optimal decision problem for a 2nd-order DPA attack:

\[
\prod_{k=0}^{N-1} \cosh(b_k + c_k) < \prod_{k=0}^{N-1} \cosh(b_k - c_k)
\]
Masking Countermeasure Conclusions

- **Without Masking Countermeasures:**
  - 1st-order DPA attack
  - In my experiments, fewer than $N = 50$ power signals are needed

- **With Masking Countermeasures:**
  - 2nd-order DPA attack
  - Attacker needs to know which points in the power signal to monitor
  - More power signals are needed: $2N^2$
  - In my experiments, many secret bits can leak with fewer than 50 signals
Summary of Motorola Labs Research

- Developed models to understand how and why power analysis attacks work
- Examined vulnerabilities in symmetric-key and public-key algorithms
- Reported on advanced attacks
- Analyzed countermeasures
Future Research

- Develop and evaluate hardware countermeasures
- Develop more secure (yet practical) software countermeasures
- Standard methods for testing implementations
Any Questions?

Contact:

Tom.Messerges@motorola.com