Introduction

- Classic cryptography views the secure problems with mathematical abstractions
- The classic cryptanalysis has had a great success and promise
  - Analyzing and quantifying crypto algorithms’ resilience against attacks
- Recently, many of the security protocols have been attacked through physical attacks
  - Exploit weaknesses in the cryptographic system hardware implementation aimed to recover the secret parameters

Side Channel Emissions

- Side-Channel attacks aim at side-channel inputs and outputs, bypassing the theoretical strength of cryptographic algorithms
- Five commonly exploited side-channel emissions:
  - Power Consumption
  - Electro-Magnetic
  - Optical
  - Timing and Delay
  - Acoustic

Hardware Targets

- Two common victims of hardware cryptanalysis are smart cards and FPGAs
  - Attacks on smart cards are applicable to any general purpose processor with a fixed bus architecture.
  - Attacks on FPGAs are also reported. FPGAs represent application specific devices with parallel computing opportunities.

Smart Cards

- Smart cards have a small processor (8bit in general) with ROM, EEPROM and a small RAM
- Eight wires connect the processor to the outside world
- Power supply: no internal batteries
- Clock: no internal clock
- Typically equipped with a shield that destroys the chip if a tampering happens
FPGAs

- FPGAs allow parallel computing
- Multiple programmable configuration bits

Attack Model

- Consider a device capable of implementing the cryptographic function
- The key is usually stored in the device and protected
- Modern cryptography is based on Kerckhoffs’s assumption → all of the data required to operate a chip is entirely hidden in the key
- Attacker only needs to extract the key

Physical Attack Phases

- Physical attacks are usually composed of two phases:
  - Interaction phase: interact with the hardware system under attack and obtain the physical characteristics of the device
  - Analysis phase: analyze the gathered information to recover the key

Principle of divide-and-conquer attack

- The divide-and-conquer(D&C) attack attempt at recovering the key by parts
- The idea is that an observed characteristic can be correlated with a partial key
  - The partial key should be small enough to enable exhaustive search
- Once a partial key is validated, the process is repeated for finding the remaining keys
- D&C attacks may be iterative or independent

Attack Classification

- Invasive vs. noninvasive attacks
- Active vs. passive attacks
  - Active attacks exploit side-channel inputs
  - Passive attacks exploit side-channel outputs
- Simple vs. differential attacks
  - Simple side-channel attacks directly map the results from a small number of traces of the side-channel to the operation of DUA
  - Differential side-channel attacks exploit the correlation between the data values being processed and the side-channel leakage

Power attacks

- Measure the circuit’s processing time and current consumption to infer what is going on inside it.
Measuring Phase

- The task is usually straightforward
  - Easy for smart cards: the energy is provided by the terminal and the current can be read
- Relatively inexpensive (<$1000) equipment can digitally sample voltage differences at high rates (1GHz++) with less than 1% error
- Device’s power consumption depends on many things, including its structure and data being processed

Power Analysis

- Monitor the device’s power consumption to deduce information about data and operation
- Summary of DES – a block cipher
  - a product cipher
  - 16 rounds iterations
    - substitutions (for confusion)
    - permutations (for diffusion)
  - Each round has a round key
    - Generated from the user-supplied key

DES Basic Structure

- Input: 64 bits (a block)
- Li/Ri – left/right half (32 bits) of the input block for iteration i – subject to substitution S and permutation P
- K - user-supplied key
- Ki - round key:
  - 56 bits used +8 unused (unused for encryption but often used for error checking)
- Output: 64 bits (a block)
- Note: Ri becomes L(i+1)
- All basic op’s are simple logical ops
  - Left shift / XOR

SPA on Modular Mul or Exp

- SPA can be used to break cryptographic implementations
- Multipliers: Involves modular multiplication – The leakage function depends on the multiplier design but strongly correlated to operand values and Hamming weights
- Exponentiators: Involves squaring operation and multiplication
- SPA Countermeasure:
  - Avoid procedures that use secret intermediates or keys for conditional branching operation

SPA on Modular Mul or Exp (cont’d)

- Modular exponentiation is often implemented by square and multiply algorithm
- Typically the square operation is implemented differently compared with the multiply (for speed purposes)
- Then, the power trace of the exponentiation can directly yields the corresponding value
- All programs involving conditional branching based on the key values are at risk!
Differential power analysis (DPA)

- SPA targets variable instruction flow
- DPA targets data-dependence
  - Different operands presents different power
- Difference between smart cards and FPGAs
  - In smart cards, one operation running at a time
    - Simple power tracing is possible
  - In FPGAs, typically parallel computations prevent visual SPA inspection → DPA

DPA

- DPA can be performed on any algorithm that has the operation $\beta = S(\alpha \oplus K)$.
  - $\alpha$ is known and $K$ is the segment key

Play the algorithm $N$ times
($100 < N < 100000$)

Input data (messages $M_i$)

Algorithm

Output (cipher texts $C_i$)

Power Consumption Curves $W_i$ (or other data channel leakage like ISM calculation)

The waveforms are captured by a scope and sent to a computer for analysis

DPA (cont’d)

Assumption: Either Plaintext or Cipher is known

What is available after acquisition?

- After data collection, what is available?
  - $N$ plain and/or cipher random texts
  - $00$ 868BE57BB63ED1E
  - $01$ 185DDOMT27590F36F
  - $02$ C031A0392DC88E16
  - $N$ corresponding power consumption waveforms

DPA (cont’d)

The bit will classify the wave $w_i$

- Hypothesis 1: bit is zero
- Hypothesis 2: bit is one

A differential trace will be calculated for each bit!

DPA (cont’d)

Partition the data and related curves into two packs, according to the selection bit value:

$M_i$ bit ($M_i$) = 0
$M_i$ bit ($M_i$) = 1

... and assign -1 to pack 0 and +1 to pack 1

Sum the signed consumption curves and normalise

$\Delta_x = \sum x_i \frac{w_i}{N_x} - \sum x_i \frac{w_i}{N_x}$
The DPA waveform with the highest peak will validate the hypothesis.

Assumption: Attacker presumes detailed knowledge of the DES

Divide-and-conquer strategy, comparing powers for different inputs
- Record large number of inputs and record the corresponding power consumption
- Start with round 15 - We have access to $R_{15}$, that entered the last round operation, since it is equal to $L_{16}$
- Take this output bit (called $M_i$) at the last round and classify the curves based on the bit
  - 6 specific bits of $R_15$ will be XOR’d with 6 bits of the key, before entering the S-box
  - By guessing the first key value, we can predict the bit $b_i$, or an arbitrary output bit of an arbitrary S-box output.

A closer look at HW Implementation Of DES

Example: DPA on DES

Attacking a secret key algorithm

DPA works thanks to the perfect prediction of the selection bit

How to break a key?
Timing attacks

- Running time of a crypto processor can be used as an information channel
- The idea was proposed by Kocher, Crypto’96
  - You put $28 in one of the pots and $10 in the other:
  
  ![Image of two pots with money]  

  - Question: Compute
    - Blue * 10 + Red * 7
    - Tell me if the result is odd or even.
    - Is your answer enough to reveal what’s in each pot?

RSA Cryptosystem

- Key generation:
  - Generate large (say, 2048-bit) primes p, q
  - Compute n=pq and \( \varphi(n) = (p-1)(q-1) \)
  - Choose small e, relatively prime to \( \varphi(n) \)
    - Typically, e=3 (may be vulnerable) or e=2^16+1=65537 (why?)
  - Compute unique d such that \( ed \equiv 1 \pmod{\varphi(n)} \)
  - Public key = (e, n); private key = (d, n)
  - Security relies on the assumption that it is difficult to factor n into p and q
- Encryption of m: \( c = m^e \mod n \)
- Decryption of c: \( c^d \mod n = (m^e)^d \mod n = m \)

How Does RSA Decryption Work?

- RSA decryption: compute \( y^d \mod n \)
- This is a modular exponentiation operation
- Naive algorithm: square and multiply

Let \( y_0 = 1 \).
For \( k = 0 \) up to \( w - 1 \):
  - If (bit \( k \) of \( x \)) is 1 then
    - Let \( R_k = (y_k \cdot y) \mod n \).
  - Else
    - Let \( R_k = y_k \).
    - Let \( y_{k+1} = R_k^2 \mod n \).
End For.
Return \((R_{w-1})\).
Kocher’s Observation

Let \( s_0 = 1 \).
For \( k = 0 \) up to \( w-1 \):
If (bit \( k \) of \( x \)) is 1 then
    Let \( R_k = (s_k \cdot y) \mod n \).
Else
    Let \( R_k = 1 \).
Let \( s_{k+1} = R_k^2 \mod n \).
EndFor.
Return \( (R_w^{-1}) \).

Outline of Kocher’s Attack

- Idea: guess some bits of the exponent and predict how long decryption will take
- If guess is correct, we will observe correlation; if incorrect, then prediction will look random
  - This is a signal detection problem, where signal is timing variation due to guessed exponent bits
  - The more bits you already know, the stronger the signal, thus easier to detect (error-correction property)
- Start by guessing a few top bits, look at correlations for each guess, pick the most promising candidate and continue

Electromagnetic Power Analysis

EMA – probe design

- Handmade solenoid
- Diameter = 150 to 500 \( \mu \)m
- Bandwidth > 100 MHz, low voltage, parasitic effects
- Good acquisition chain required, but no Faraday cage
- Sampling at 10Hz

EMA signal

- Raw signals (TIA: transfer into accumulator instruction)
  - Power is less noisy
  - But EM signatures are sharper

Spatial Positioning

- Horizontal cartography OFF plane
  - to pinpoint instruction related areas
  - better if automated
Advantage of EMA versus PA
- Local information more “data correlated”
- EMA bypasses current smoothers
- EMA goes through some HW countermeasures: power shields, randomized logic

Drawbacks
- Experimentally more complicated
- Geometrical scanning can be tedious
- Low level and noisy signals (decapsulation required)

General Countermeasures
- Hiding — reduce the SNR by either increasing the noise or reducing the signal
  - Noise Generators
  - Balanced Logic Styles
    - DPR (dual power rail)
- Asynchronous Logic
  - PA resistive (dual rail or 1-of-n signal encoding logic)
  - Less power emission (no clock)
  - Should handle the calculation time dependency issue

General Countermeasures (cont’d)
- Hiding — reduce the SNR by either increasing the noise or reducing the signal
  - Low Power Design
    - Reducing the overall power consumption
    - Meanwhile one should be careful about other side channels
  - Shielding
    - For power using regulators and such
    - To damp the EM or acoustic emissions

General Countermeasures (cont’d)
- Masking/Blinding — remove the correlation between the input data and the side-channel emissions from intermediate nodes in the functional block
  - Masking input before going to cryptographic oracle
  - Unmasking the outputs
  - To de-correlate the input data and intermediate values
General Countermeasures (cont’d)

- Design Partitioning — separate regions of the chip that operate on plaintext from regions that operate on ciphertext
  - Preventing to have the coupling effect of one part (secret one) into another part.
  - Separating
    - Clock logic (DLL)
    - Power logic
    - Test logic (scan chains, BISTs)

- Physical Security and Anti-Tamper — denial of proximity, access, and possession
  - By denial of proximity, access and possession, the possibility of side channel information reduces
    - E.g., NSA rules out that around 200 feet from any crypto center should be physically protected and watched.
    - Acoustic shielding of chips might be required due to the use of long term microphones technologies.

EMA Countermeasures

- Software (crypto routines)
  - Coding techniques
  - Same as anti DPA/SPA (data whitening…)

- Hardware (chip designers)
  - Confine the radiation (metal layer)
  - Blur the radiation (e.g. by an active emitting grid)
  - Reduce the radiation (technology trends to shrinking)
  - Cancel the radiation (dual logic)

Side-Channel Attacks and Countermeasures for Embedded Microcontrollers

Source of side-channel leakage in a microcontroller

- Memory-store instructions
- Memory-load instructions
- Arithmetic instructions
- Control-flow instructions

Side-Channel Attacks on Microcontrollers

Objective: retrieve the internal secret key $k^*$ of a crypto-algorithm

- The leakage caused by $\nu$ is a function of the key value $k^*$, and it can be expressed as follows:
  $$L(k^*) = f_{\nu}(p) + \varepsilon$$

The function $f_{\nu}$ is dependent on the crypto-algorithm as well as on the nature of the implementation in hardware and software. The error $\varepsilon$ is an independent noise variable.
Correlation Power Analysis

- Two important aspects of a practical CPA:
  - The selection of the power model
    The power model is chosen so that it has a dependency on a part of the secret key. A good candidate is the output of the substitution step.
  - The definition of the attack success metric
    Measurements to Disclosure (MTD): the more measurements that are required to successfully attack a cryptographic design with side-channel analysis, the more secure that design is.

Practical Hypothesis Tests

- An example of 256 correlation coefficient traces. Around time 100 us, the black trace which corresponds to the correct key byte emerges from all the other 255 traces.
Side Channel Countermeasures for Microcontrollers

- Two different kinds of countermeasures:
  - Algorithm-Level Countermeasures
    Transform the C program so that the generation of dangerous side-channel leakage is avoided.
  - Architecture-Level Countermeasures
    Create a better microcontroller, for example using special circuit techniques, so that no side-channel leakage is generated.

---

Dual Rail Precharge

- A CMOS standard NAND has data-dependent power dissipation;
- A DRP NAND gate has a data-independent power dissipation
- DRP requires the execution of the direct and complementary data paths in parallel.

---

VSC: Porting DRP into software

- Concept of balanced processor and VSC programming;
- The balanced processor does not show side-channel leakage.
- The power dissipation from the direct operation always has a complementary counterpart from the complementary operation. The sum of these two is a constant.

---

References