Increasing the Robustness of the Montgomery *kP*-Algorithm against SCA by Modifying its Initialization

Alpírez Bock, Estuardo
Dyka, Zoya
Langendörfer, Peter

innovations
for high
performance
microelectronics

09th June 2016

International Conference on Information Technology and Communication Security

Member of Leibniz Association

Outline

Side Channel Analysis Attacks

SCA attacks are *passive* physical attacks based on the observation of a cryptographic device during its execution of cryptographic operations.

Parameters observed:

- Power consumption (power analysis)
- Electromagnetic radiation (EM analysis)
- Execution times (timing attacks)

Active attacks influence the behaviour of the device being analysed

e.g. through fault injection (FS analysis)

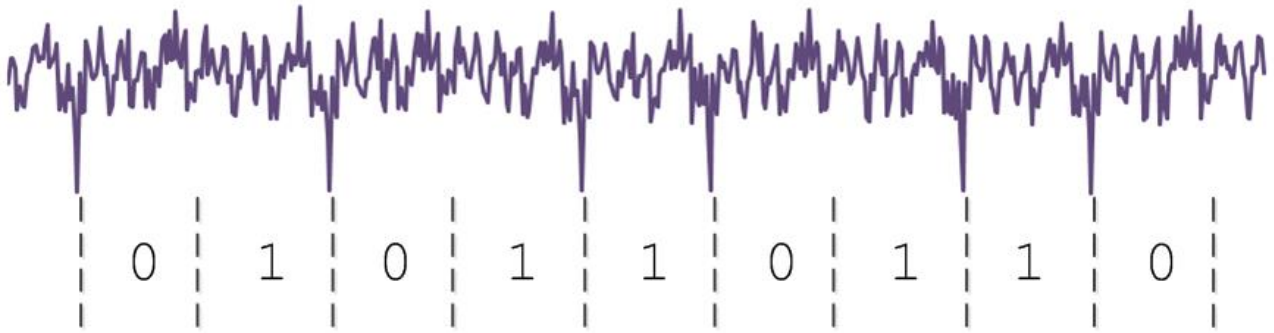
Elliptic Curve Cryptography

ECC is an asymmetric cryptography approach based on elliptic curves (*E*) over Galois Fields (*GF*).

- 99% of a decryption operation in ECC consists of the **EC point multiplication** *kP* (also *kP*-operation), where *k* is the private key.

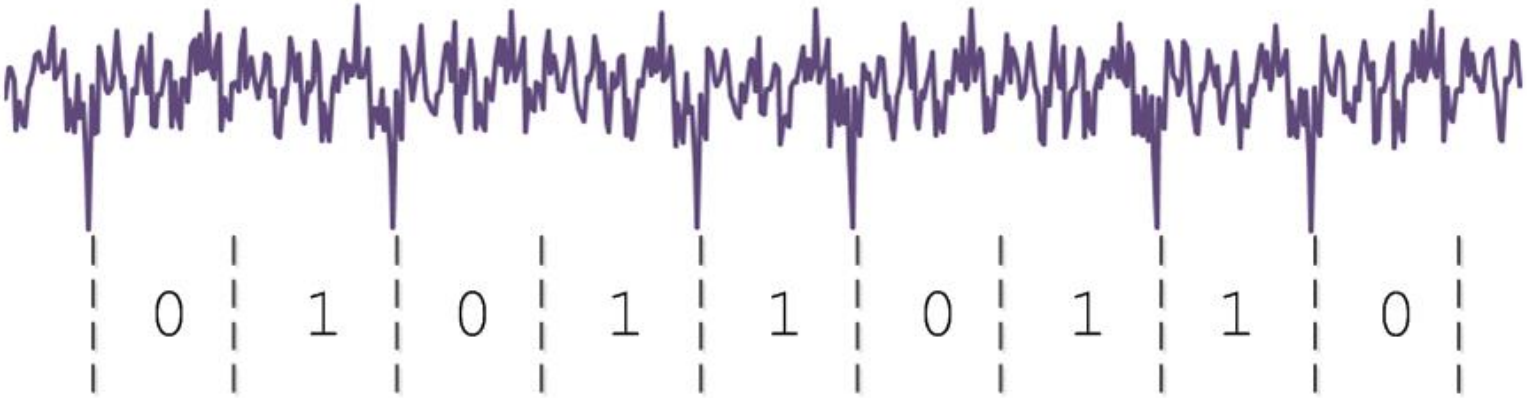
$$kP = \underbrace{P + \cdots + P}_{k}$$

- The *kP*-operation is a bitwise processing of *k*. Power traces of *kP* can be analysed for performing a key extraction.

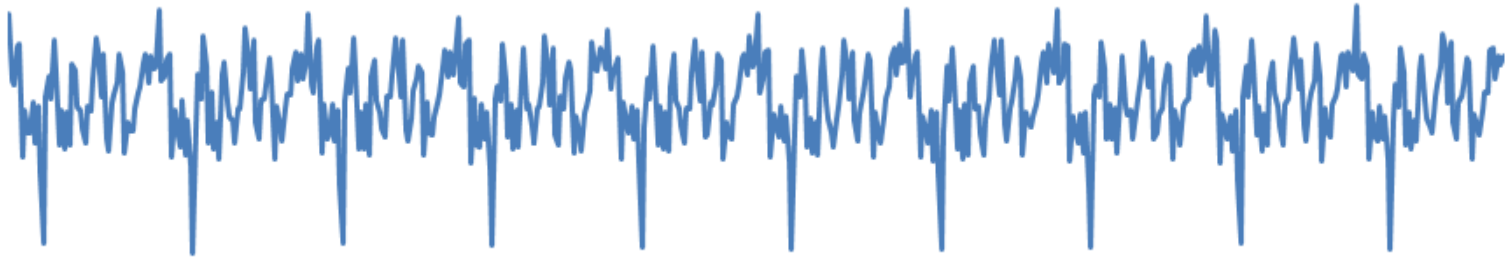


Power Analysis

- SPA



- DPA



Montgomery *kP*-Algorithm using L-D Projective Coordinates

Input: $k = (k_{l-1}, ..., k_1, k_0)_2$ with $k_{l-1} = 1, P = (x, y) \in E(GF(2^m))$
Output: $kP = (x_1, y_1)$

1: $X_1 \leftarrow x$, $Z_1 \leftarrow 1$, $X_2 \leftarrow x^4 + b$, $Z_2 \leftarrow x^2$
2: **for** i from $l-2$ downto 0 **do**
3: **if** $k_i = 1$ **then**
4: $T \leftarrow Z_1$, $Z_1 \leftarrow (X_1 Z_2 + X_2 Z_1)^2$, $X_1 \leftarrow xZ_1 + X_1 X_2 T Z_2$
5: $T \leftarrow X_2$, $X_2 \leftarrow X_2^4 + bZ_2^4$, $Z_2 \leftarrow T^2 Z_2^2$
6: **else**
7: $T \leftarrow Z_2$, $Z_2 \leftarrow (X_2 Z_1 + X_1 Z_2)^2$, $X_2 \leftarrow xZ_2 + X_1 X_2 T Z_1$
8: $T \leftarrow X_1$, $X_1 \leftarrow X_1^4 + bZ_1^4$, $Z_1 \leftarrow T^2 Z_1^2$
9: **end if**
10: **end for**
11: $x_1 \leftarrow X_1 / Z_1$
12: $y_1 \leftarrow y + (x + x_1)[X_1 + xZ_1)(X_2 + xZ_2) + (x^2 + y)(Z_1 Z_2)]/(xZ_1 Z_2)$
13: **return** $((x_1, y_1))$

Power Trace of an Efficient Implementation of the Montgomery *kP*-Algorithm

Montgomery *kP*-Algorithm using L-D Projective Coordinates

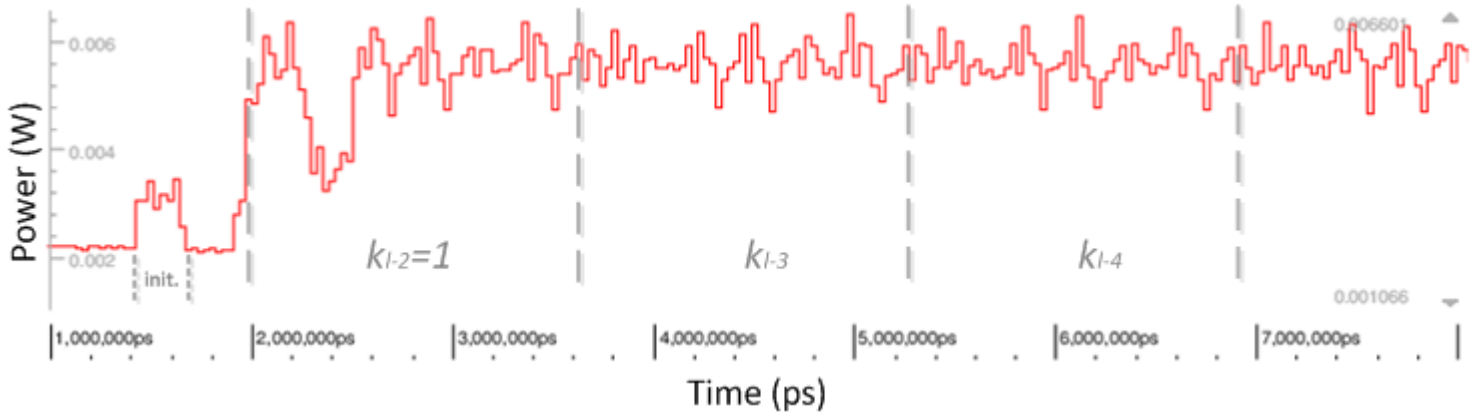
Input: $k = (k_{l-1}, ..., k_1, k_0)_2$ with $k_{l-1} = 1, P = (x, y) \in E(GF(2^m))$
Output: $kP = (x_1, y_1)$

1: $X_1 \leftarrow x,\ Z_1 \leftarrow 1,\ X_2 \leftarrow x^4 + b,\ Z_2 \leftarrow x^2$
2: **for** i from $l-2$ downto 0 **do**
3: **if** $k_i = 1$ **then**
4: $T \leftarrow Z_1,\ Z_1 \leftarrow (X_1 Z_2 + X_2 Z_1)^2,\ X_1 \leftarrow xZ_1 + X_1 X_2 T Z_2$
5: $T \leftarrow X_2,\ X_2 \leftarrow X_2^4 + bZ_2^4,\ Z_2 \leftarrow T^2 Z_2^2$
6: **else**
7: $T \leftarrow Z_2,\ Z_2 \leftarrow (X_2 Z_1 + X_1 Z_2)^2,\ X_2 \leftarrow xZ_2 + X_1 X_2 T Z_1$
8: $T \leftarrow X_1,\ X_1 \leftarrow X_1^4 + bZ_1^4,\ Z_1 \leftarrow T^2 Z_1^2$
9: **end if**
10: **end for**
11: $x_1 \leftarrow X_1/Z_1$
12: $y_1 \leftarrow y + (x + x_1)[X_1 + xZ_1)(X_2 + xZ_2) + (x^2 + y)(Z_1 Z_2)]/(xZ_1 Z_2)$
13: **return** $((x_1, y_1))$

First Iteration of the Loop (if $k_{l-2} = 1$)

if $k_{l-2} = 1$

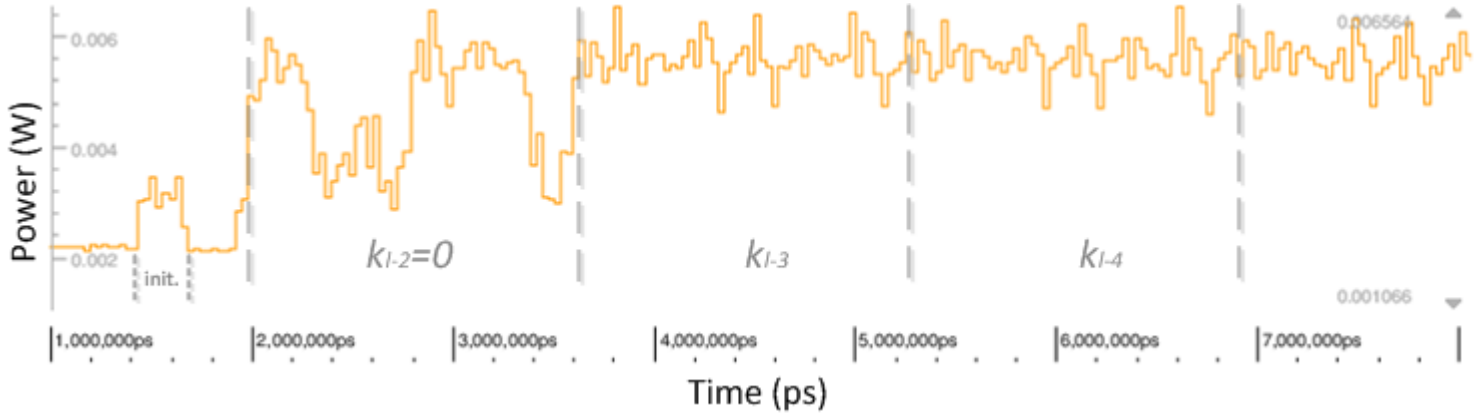
$$T \leftarrow 1, Z_1 \leftarrow (X_1 Z_2 + X_2 \cdot 1)^2, X_1 \leftarrow xZ_1 + (X_1 Z_2)(X_2 \cdot 1)$$
$$T \leftarrow X_2, X_2 \leftarrow (X_2^2)^2 + b(Z_2^2)^2, Z_2 \leftarrow T^2 Z_2^2$$



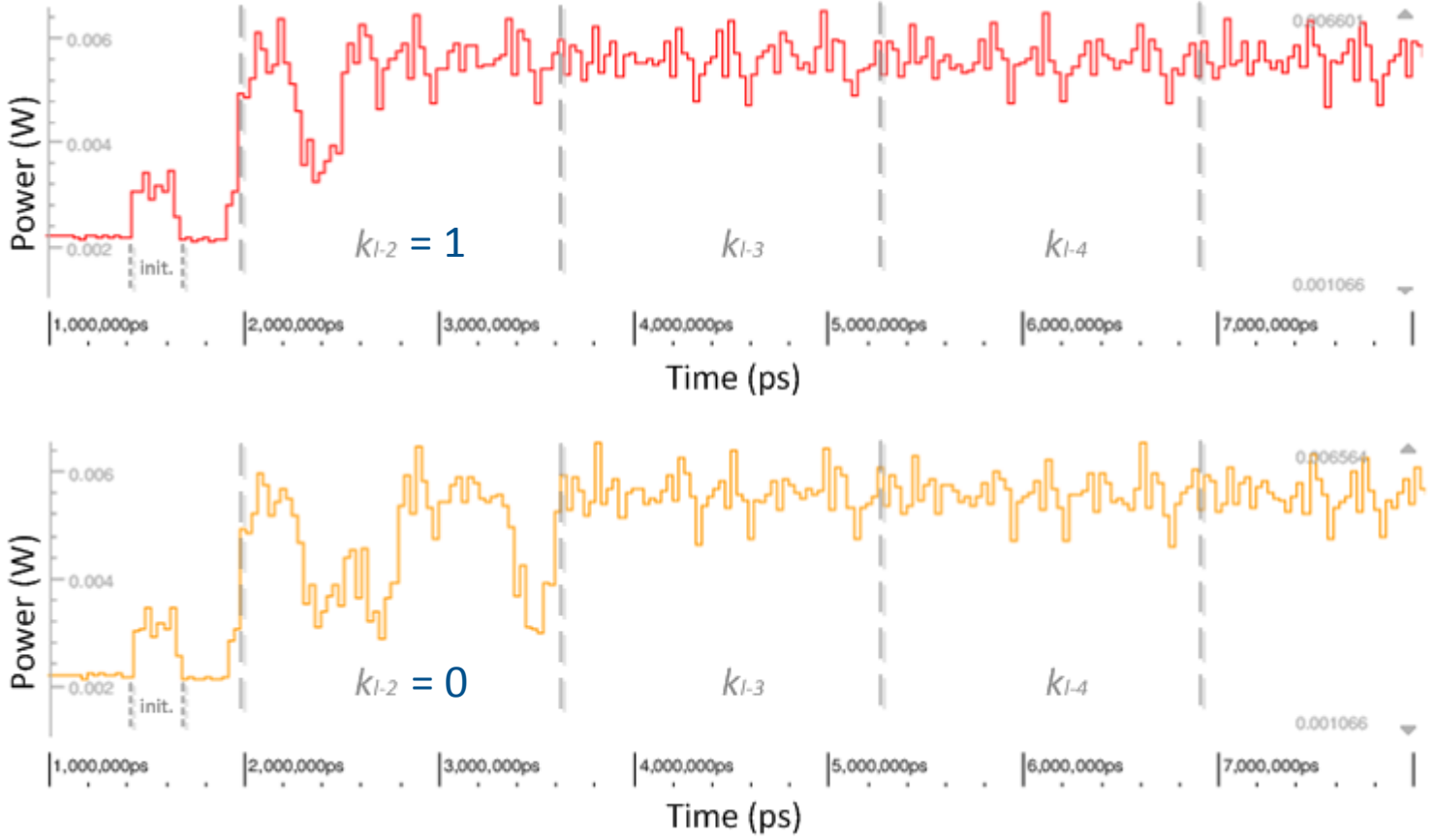
First Iteration of the Loop (if $k_{l-2} = 0$)

if $k_{l-2} = 0$

$$T \leftarrow Z_2, Z_2 \leftarrow (X_2 \cdot 1 + X_1 Z_2)^2, X_2 \leftarrow xZ_2 + (X_1 T)(X_2 \cdot 1)$$
$$T \leftarrow X_1, X_1 \leftarrow (X_1^2)^2 + b(1^2)^2, Z_1 \leftarrow T^2 \cdot 1^2$$



Observation: extraction of the first bit is possible through SPA



- The form of the first slot can reveal information about the implemented algorithm

Proposed Solution

k_{l-2} is processed outside of the main loop, with a different (shorter) operation flow, in order to avoid performing calculations with an operand with value 1.

The initialization phase is shortened as well:

- Original initialization phase

$$X_1 \leftarrow x, Z_1 \leftarrow 1, X_2 \leftarrow x^4 + b, Z_2 \leftarrow x^2$$

- New initialization phase

$$X_1 \leftarrow x, X_2 \leftarrow x^4 + b, Z_2 \leftarrow x^2$$

Processing of Bit k_{l-2} outside of the Loop

if $k_{l-2} = 1$ **then**

$T \leftarrow Z_2,\ Z_1 \leftarrow (X_1 Z_2 + X_2)^2,\ X_1 \leftarrow X_1 Z_2 X_2 + Z_1 x$

$T \leftarrow X_2,\ U \leftarrow bZ_2^4,\ X_2 \leftarrow X_2^4 + U,\ U \leftarrow TZ_2,\ Z_2 \leftarrow U^2$

else

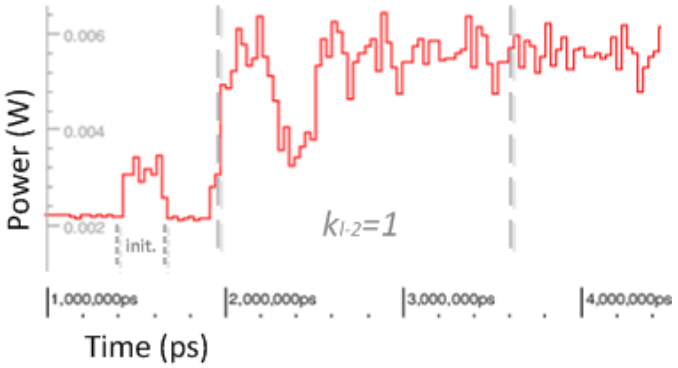
$T \leftarrow Z_2,\ Z_2 \leftarrow (X_1 Z_2 + X_2)^2,\ X_2 \leftarrow X_1 X_2 T + Z_2 x$

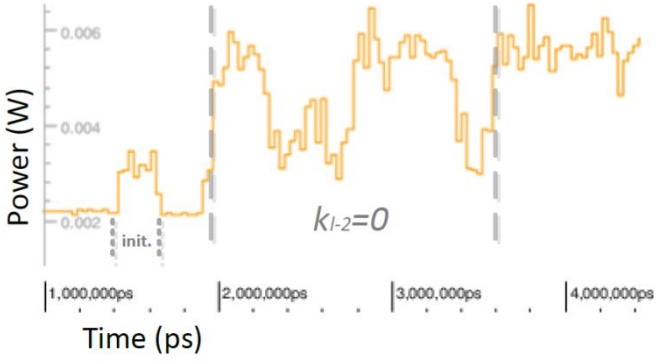
$T \leftarrow X_1,\ U \leftarrow bX_2^4,\ X_1 \leftarrow X_1^4 + b,\ U \leftarrow TX_2,\ Z_1 \leftarrow T^2$

- Only 5 multiplications are calculated for bit k_{l-2}
- Dummy operations need to be performed in both cases in order to reach a balanced operation flow for this bit

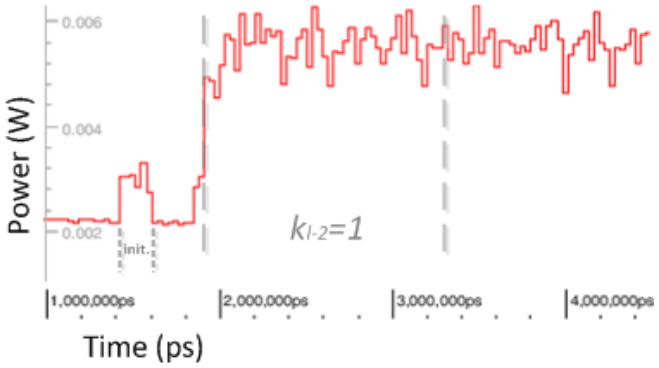
PTs of the *kP*-Operation: *kP*-Algorithm with and without our Modifications

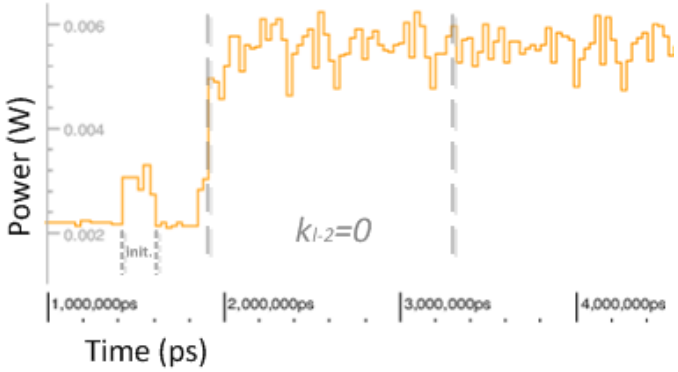
Before the modifications





After the modifications





Comparison of our Implementations of the *kP*-Algorithm with and without Modifications (IHP 130 nm technology)

Montgomery Implementation			Before modifications	After modifications
Initialization phase	Cycles		7	5
	Energy		0.63 nJ	0.46 nJ
Processing of bit k_{l-2}	Cycles		54	45
	Energy	k_{l-2}=1	8.60 nJ	7.45 nJ
		k_{l-2}=0	7.60 nJ	7.45 nJ
Extraction of bit k_{l-2} with SPA			Yes	No
Implementation details revealed			Yes	No
kP	Cycles		12915	12904
	Energy		2.10 µJ	2.09 µJ
	Area		0.2745 mm²	0.2748 mm²

Conclusions

- A vulnerability in the initializaiton phase of the Montgomery *kP*-Algorithm using L-D projective coordinates was identified and countermeasured.
- Protection against SPA for the key bit k_{l-2} has been achieved through this countermeasure.
- The robustness against further PA attacks, such as DPA and template attacks has been increased.
- Our modifications in the initialization phase of the algorithm did not imply any additional costs

Thank you for your attention!
Mulţumesc!

Estuardo Alpírez Bock

IHP – Innovations for High Performance Microelectronics
Im Technologiepark 25
15236 Frankfurt (Oder)
Germany
Phone: +49 (0) 335 5625 522
Fax: +49 (0) 335 5625
Email: alpirez@ihp-microelectronics.com

www.ihp-microelectronics.com

innovations
for high
performance
microelectronics

Member of
Leibniz Association

Initialisierungsphase

$$X_1 \leftarrow x, Z_1 \leftarrow 1, X_2 \leftarrow x^4 + b, Z_2 \leftarrow x^2$$

- Register Z_1 wird mit dem Wert 1 initialisiert

Physikalische Angriffe

SCA (engl. Side Chanel Analyses) sind *passive* Angriffe und basieren auf der Beobachtung eines kryptographischen Gerätes während der Ausführung kryptographischer Operationen.

Beobachtet wird:

- Stromverbrauch (Power Analyse)
- Elektromagnetische Strahlung (EM Analyse)
- Ausführungszeit (Timing Attacks)

SCA kann durch einfache Visualisierung, Anwendung statistischer Methoden oder Herstellung von Templates durchgeführt werden.

Aktive Angriffe beeinflussen das Verhalten des analysierten Gerätes

z. B. durch Fehlerinjektion (FS Analyse) zur Herstellung von Templates

Gliederung

Gliederung

Gliederung

Physikalische Angriffe

- Passive Angriffe (SCA (engl. Side Chanel Analyses))
  - Basieren auf der Analyse der physikalischen Parametern bzw. Phänomenen, die während Ausführung kryptographischer Operationen beobachtet und gemessen werden können, z.B.
    - Stromverbrauch (Power Analyse)
    - Elektromagnetische Strahlung (EM Analyse)
    - Ausführungszeit (Timing Attacks)
    - Photoemission
    - Etc.
  - Die Messergebnisse können analysiert werden durch/mit
    - Visualisierung (SPA, SEMA)
    - mit Hilfe von statistischen Methoden (DPA, DEMA)
    - Herstellung von Templates
- Aktive Angriffe
  - beeinflussen das Verhalten des analysierten Gerätes (z.B durch Fehlerinjektion)
    - Herstellung von Templates möglich

Gliederung

First Iteration of the Loop

if $k_{l-2} = 1$

$$T \leftarrow 1, Z_1 \leftarrow (X_1 Z_2 + X_2 \cdot 1)^2, X_1 \leftarrow xZ_1 + (X_1 Z_2)(X_2 \cdot 1)$$
$$T \leftarrow X_2, X_2 \leftarrow (X_2^2)^2 + b(Z_2^2)^2, Z_2 \leftarrow T^2 Z_2^2$$

if $k_{l-2} = 0$

$$T \leftarrow Z_2, Z_2 \leftarrow (X_2 \cdot 1 + X_1 Z_2)^2, X_2 \leftarrow xZ_2 + (X_1 T)(X_2 \cdot 1)$$
$$T \leftarrow X_1, X_1 \leftarrow (X_1^2)^2 + b(1^2)^2, Z_1 \leftarrow T^2 \cdot 1^2$$

Proposed Solution

k_{l-2} is processed outside of the main loop, with a different (shorter) operation flow, in order to avoid performing calculations with an operand with value 1.

Shortened initialization phase:

$$X_1 \leftarrow x, X_2 \leftarrow x^4 + b, Z_2 \leftarrow x^2$$

- Register Z_1 is not initialized
- No register is initialized with the value 1