CHES 2013

A New Model for Error-Tolerant Side-Channel Cube Attacks

Zhenqi Li, Bin Zhang, Junfeng Fan and Ingrid Verbauwhede

Institute of Software, Chinese Academy of Sciences, China
State Key Laboratory of Information Security, IIE, China
Katholieke Universiteit Leuven, ESAT SCD/COSIC

Outline

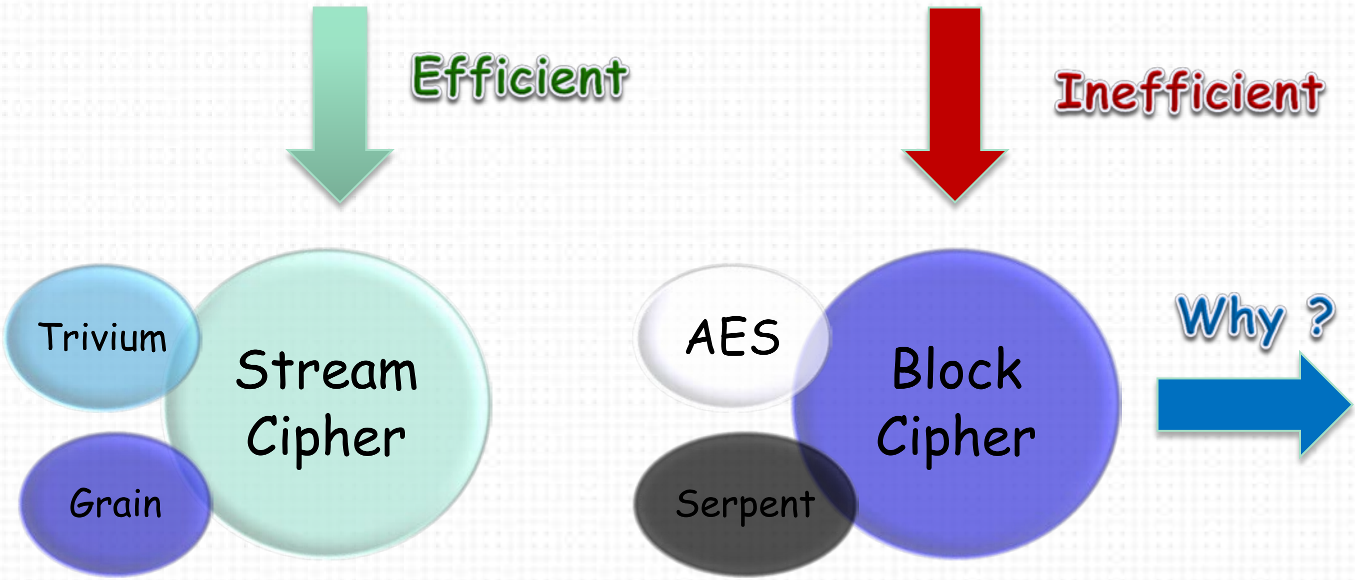
- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Cube attack

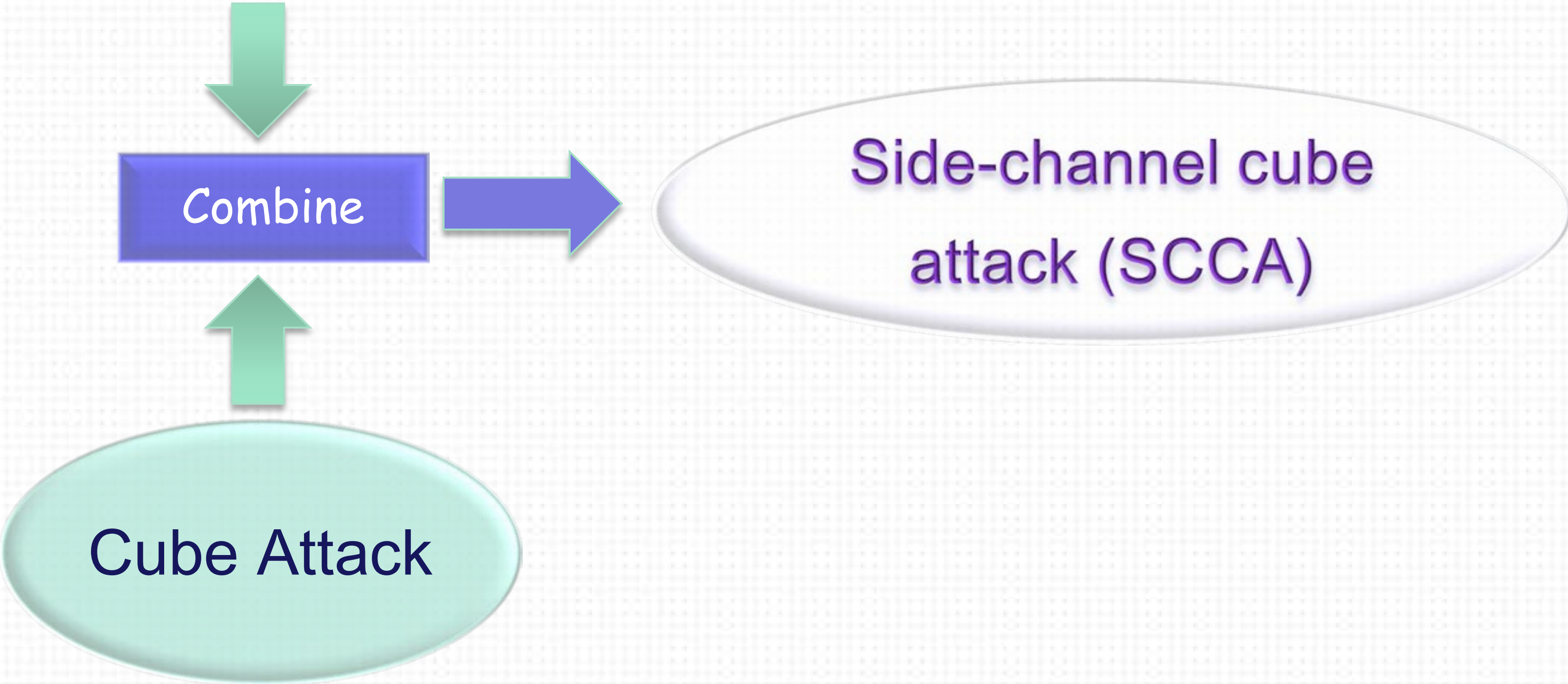
- A new branch of algebraic attacks.
- Formally proposed by Dinur and Shamir (EUROCRYPT 2009).
- A generic key extraction attack.



- Highly non-linear round function
- Iterate many times
- High algebraic complexity for any output bit

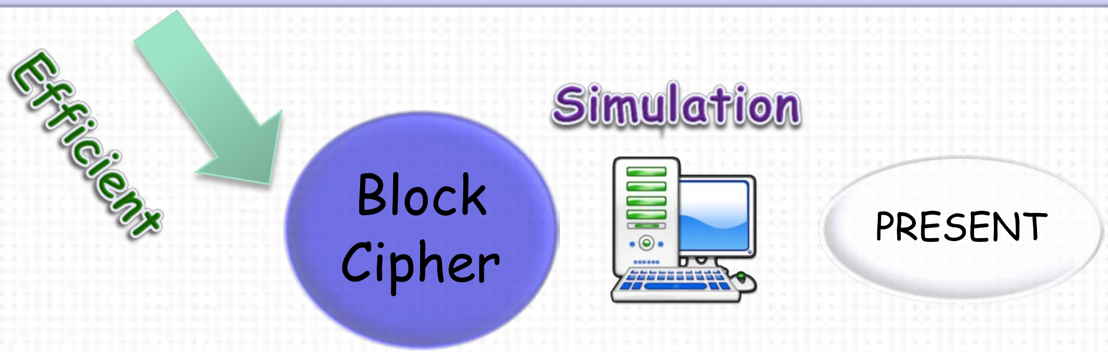
Side-channel attacks

- The attackers can learn some intermediate leakage.
- The leakage contains key related information.
- Power analysis, EM analysis, Timing…



Side-channel cube attacks

- Plaintexts, ciphertexts, intermediate variables (i.e., state registers).
- Learn the value of a single wire or register, ideal for probing attack.
- Low-degree polynomials on intermediate variables.
- Apply cube attack to those leakage. **Main challenge: measurement errors.**



Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Cube attack

- Off-line phase: finding appropriate cubes, performed once per cryptosystem.
- On-line phase: Deduce a group of linear equations and solve it to retrieve key.

Consider a block cipher: $(c_1,\dots,c_m)=E(k_1,\dots,k_n,v_1,\dots,v_m)$

$$c_i = p(k_1,\dots,k_n,v_1,\dots,v_m)$$

$$p(k_1,\dots,k_n,v_1,\dots,v_m)=t_I\cdot p_{S(I)}+q(k_1,\dots,k_n,v_1,\dots,v_m)$$

Where $t_I=\prod_{i\in I}v_i$, t_I is called a maxterm of p when $\deg(p_{S(I)})\equiv 1$.

I is called a cube of p .

$$\sum_{I\in\{0,1\}^d} p(k_1,\dots,k_n,v_1,\dots,v_m)=p_{S(I)}$$ (cf. Theorem 1, [8])

Preliminaries & Notations

A toy example:

$$p(k_1,k_2,k_3,v_1,v_2,v_3)=v_2v_3k_1+v_2v_3k_2+v_1v_2v_3+v_1k_2k_3+k_2k_3+v_3+k_1+1$$
$$=v_2v_3(k_1+k_2+v_1)+(v_1k_2k_3+k_2k_3+v_3+k_1+1)$$

where $I=\{2,3\},\ t_I=v_2v_3,\ p_{S(I)}=k_1+k_2+v_1,$

$$q(k_1,k_2,k_3,v_1,v_2,v_3)=(v_1k_2k_3+k_2k_3+v_3+k_1+1)$$

The cube size is d=2, let $C_I=\{\tau_1,\tau_2,\tau_3,\tau_4\}$ and

$$\tau_1=[k_1,k_2,k_3,v_1,0,0],\quad \tau_2=[k_1,k_2,k_3,v_1,0,1],$$
$$\tau_3=[k_1,k_2,k_3,v_1,1,0],\quad \tau_4=[k_1,k_2,k_3,v_1,1,1].$$

It is easy to verify that

$$\sum_{I\in\{0,1\}^2} p = p_{|\tau_1}+p_{|\tau_2}+p_{|\tau_3}+p_{|\tau_4}=k_1+k_2+v_1=p_{S(I)}$$

Off-line phase: Find maxterms equations as many as possible.
On-line phase: Solve those maxterm equations to retrieve key.

Preliminaries & Notations

- SCCA targets on the intermediate variables, thus the evaluation of polynomial p is obtained through side-channel leakage with noise.
- Dinur and Shamir use error correction code to remove noise (DS model)
- Each measurement: 0,1 or $\perp$, $\perp$ means unreliable measurement. The Attacker assigns a new variable y_i to $\perp$.

As in the toy example:

$$k_1 + k_2 + v_1 = p_{|\tau_1} + \perp + p_{|\tau_3} + p_{|\tau_4}.$$

A new variable induced:

$$k_1 + k_2 + v_1 = p_{|\tau_1} + y_i + p_{|\tau_3} + p_{|\tau_4}.$$

- Each cube may introduce new variables, thus more equations and more measurements are required to solve the system.
- Assumption of DS model: some of the measurements must be error-free. Very challenging in real-life attacks.

Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

A New Model Based on BSC

- Consider a SCCA model that can handle errors in each measurement.
- Suppose L maxterm equations are derived

$$\begin{cases} l_1 : a_1^1 k_1 + a_1^2 k_2 + \ldots + a_1^n k_n = b_1 \\ l_2 : a_2^1 k_1 + a_2^2 k_2 + \ldots + a_2^n k_n = b_2 \\ \qquad \vdots \\ l_L : a_L^1 k_1 + a_L^2 k_2 + \ldots + a_L^n k_n = b_L \end{cases}$$

Where $b_i = \sum_{\tau \in C_i} p_{|\tau}$, $p_{|\tau}$ is obtained through measurement.

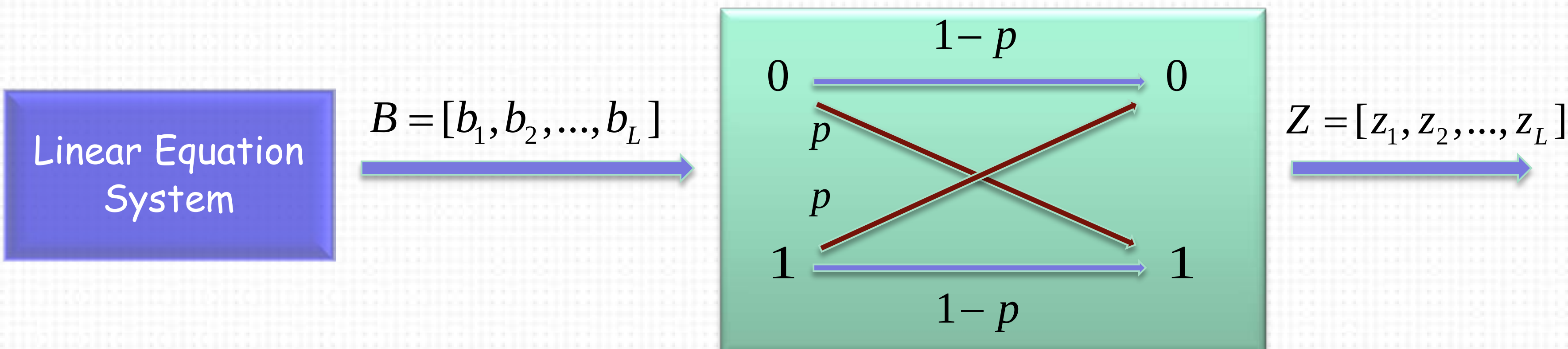
- Ideally, the measurement is error-free, the attacker obtains correct sequence $B = [b_1, b_2, \ldots, b_L]$.
- In reality, due to the measurement errors, $Z = [z_1, z_2, \ldots, z_L]$ is obtained.

A New Model Based on BSC

- q : the probability that the measurement returns a wrong bit.
- Assume $q < 1/2$ and $1-q = 1/2 + \mu$, $\mu = 0$ means a random guess.
- Since $b_i = \sum_{\tau \in C_i} p_{|\tau}$, $C_i = 2^{\bar{d}}$ and each measurement can be treated

As an independent event, according to piling-up lamma:

$$\Pr\{b_i = z_i\} \square 1 - p = 1/2 + 2^{t-1}\mu^t .$$



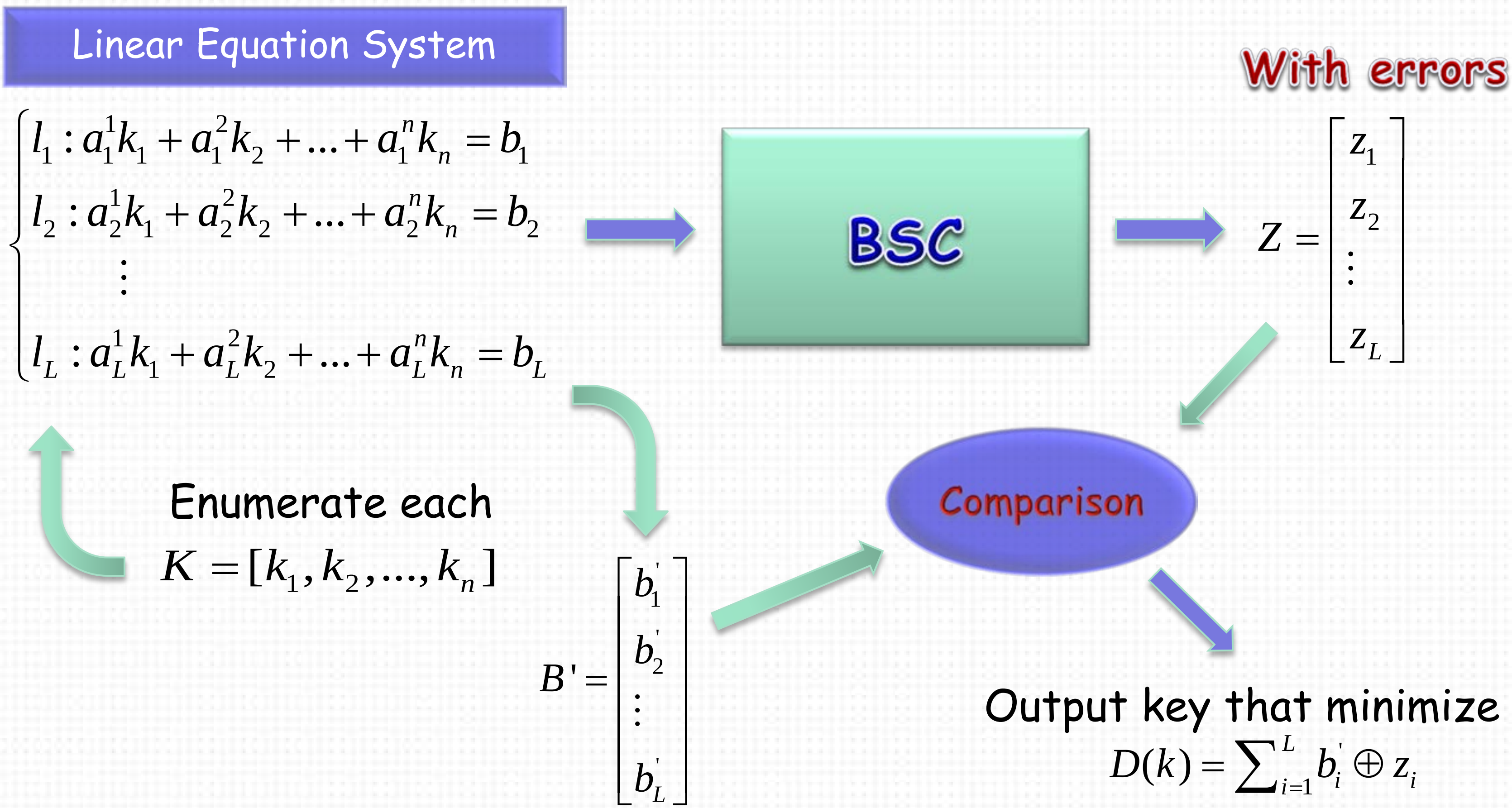
- $Z = [z_1, z_2, ..., z_L]$: received channel output.
- $B = [b_1, b_2, ..., b_L]$: codeword from an [L,n] linear block code.

Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Decoding Algorithm

•Maximum Likelihood decoding-ML decoding is adopted. Exhaustively search all the codewords of [L,n]-code. $O(2^n \cdot n / C(p))$



Decoding Algorithm

- n/L < C(p) to ensure the decoding success probability.

  $C(p)=\varepsilon^2\cdot 2/(\ln(2))$ and $p=1/2-\varepsilon$.

- When $L=l_0\approx 0.35\cdot n\cdot\varepsilon^{-2}$ the success probability approaches 50%.
- When $L=2l_0\approx 0.7\cdot n\cdot\varepsilon^{-2}$ the success probability approaches 1.

Theorem 1

$$q\le\frac{1}{2}\cdot(1-(\frac{0.35\cdot n}{L})^{\frac{1}{2t}}\cdot 2^{\frac{1}{t}})$$

where $t=2^{\bar{d}}$ and $\bar{d}$ is the average cube size.

Decoding Algorithm

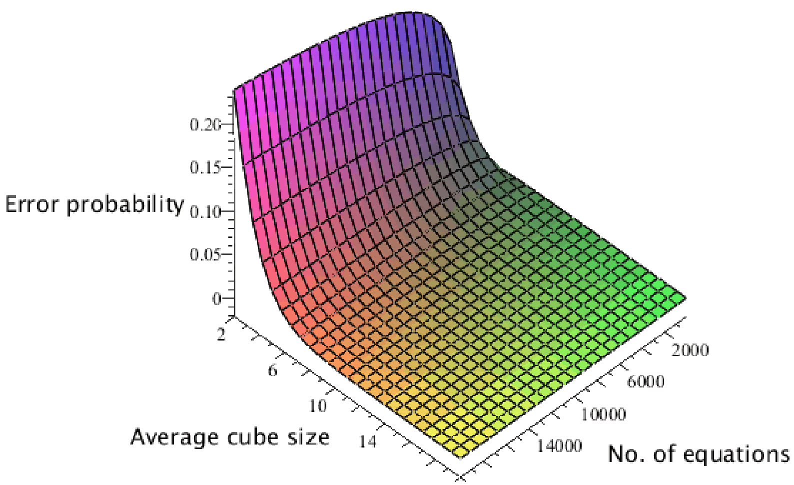


Fig.1. Error probability q as a function of d and L (Given n=80)

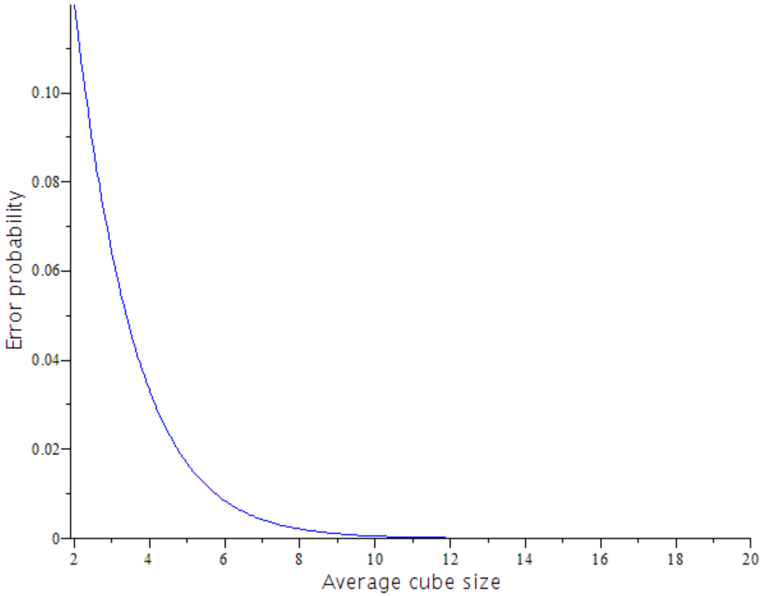


Fig.2. Error probability q as a function of d (Given L=1000, n=80)

- Error probability q exponentially decreased when cube size d increases.

Decoding Algorithm

Scenario I: when L is small

- The success probability of decoding can not be ensured.
- Store a candidate key list instead of a single key.

Scenario II: when n is big

- ML-decoding has a time complexity of 2^n .
- Use divide and conquer strategy, divide the key set into different groups and apply ML-decoding in each group.

Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Experiments and Results

PRESENT (ISO/IEC 29192-2)

- A standardized round based lightweight block cipher.
- Proposed by Bogdanov et al (CHES 2007). A cipher with SPN structure.
- Previous results of cube attacks [19,32,27] assume error-free.

```
generateRoundKeys()
for i = 1 to 31 do
    addRoundKey(STATE,K_i)
    sBoxLayer(STATE)
    pLayer(STATE)
end for
addRoundKey(STATE,K_32)
```

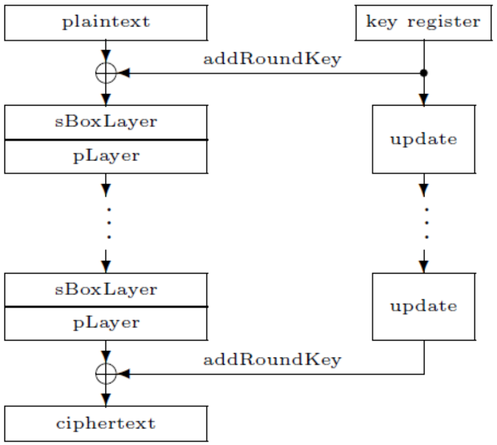
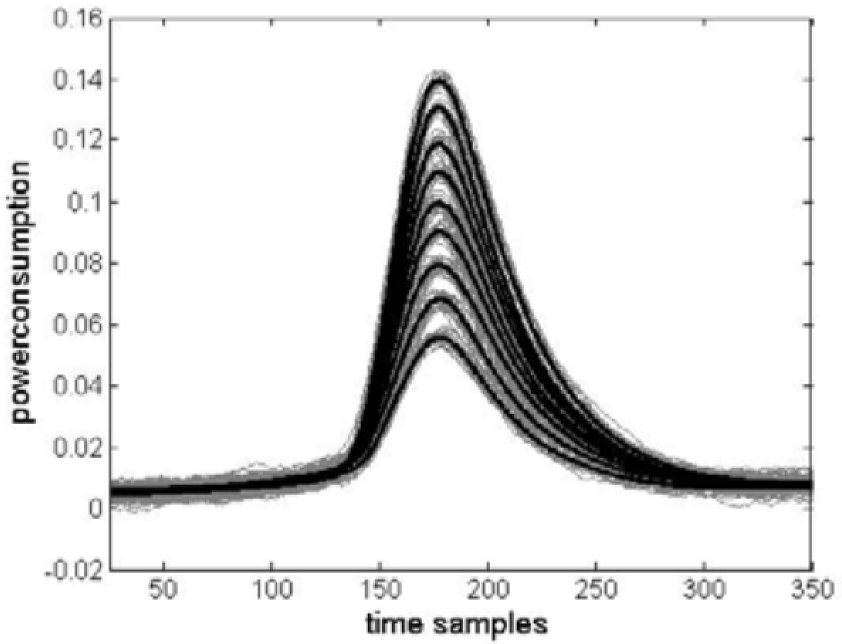


Fig.3. A top-level algorithmic description of PRESENT

Experiments and Results

•Assume: PRESENT is implemented on a 8-bit processor.

•HWL: Hamming weight leakage (state variables are loaded from memory to ALU.)



Renauld et al, CHES 2009

Experiments and Results

- Simulation on the first round:
  - Derive all the possible cubes from the LSB leakage of 8 bytes state.
  - Apply off-line phase to obtain hundreds of maxterm euqations.

Class	State bytes	Key variables	No. of maxterm equations	Average cube size
Class I	byte[1,3,5,7]	$k_{17}, k_{18}, ..., k_{48}$	150	1.90
Class II	byte[2,4,6,8]	$k_{49}, k_{50}, ..., k_{80}$	152	1.89

Divide and conquer

Group	[L,n]	Key bits	Overlapping bits
G1	[93, 20]	$k_{17}, k_{18}, ..., k_{36}$	4 with G2
G2	[95, 20]	$k_{33}, k_{34}, ..., k_{52}$	4 with G1, 4 with G3
G3	[95, 20]	$k_{49}, k_{50}, ..., k_{68}$	4 with G2, 4 with G4
G4	[76, 26]	$k_{65}, k_{66}, ..., k_{80}$	4 with G3

Experiments and Results

- The whole attack contains two phases:
  - Decoding in each group: $\sum_{i=1}^{m} t_i, t_i = 2^{n_i}$ key trials.
  - Verification phase: $Q(T) = T^m / 2^r$, T denotes the size of candidate list. r denotes the reduction factor (overlapping bits).

Leakage position	Time	Data (measurement)	r	Success probability	Error probability
LSB	$2^{21.6}$	$2^{10.2}$	12	50.1%	19.4%

Table.1. ET-SCCA on the first round

Leakage position	Time	Data (measurement)	r	Success probability	Error probability
LSB	$2^{20.6}$	$2^{18.9}$	9	61.1%	0.6%
2nd LSB	$2^{21.6}$	$2^{23.1}$	12	54.1%	0.4%

Table.2. ET-SCCA on the second round

- The error tolerance level is very low in the second round, since the cube size is relatively bigger.

Outline

- Introduction
- Preliminaries & Notations
- A New Model Based on BSC (Binary Symmetric Channel)
- Decoding Algorithms
- Experiments & Results
- Conclusion

Conclusion

- This paper considers a side-channel cube attack that can handle errors in each measurement and transform the key recover problem to the coding problem based on BSC.

- Divide and conquer strategy and list decoding technique are adopted to lower the decoding time complexity and enhance the success probability

- We simulate the attack model on PRESENT and the best result show that given about $2^{10.2}$ measurements, each with an error probability of 19.4%, it achieves 50.1% of success rate for the key recovery.

- Some open problems:
  - How to select the best target bit and find more maxterm euqations ?
  - Can side-channel cube attacks break masked implementations?
  - How to increase the error tolerance efficiently?
  - How to speed up the decoding process further (sparse structure of encoding matrix)?

Thank You