An Optimized Cross Correlation Power Attack of Message Blinding Exponentiation Algorithms

WAN Wunan[1]*, YANG Wei[1], CHEN Jun[2]

[1] Information Security Engineering College, Chengdu University of Information Technology, Chengdu 610225, China
[2] Institute of Applied Cryptography, Chengdu University of Information Technology, Chengdu 610225, China

Abstract: The message blinding method is the most efficient and secure countermeasure against first-order differential power analysis(DPA). Although cross correlation attacks(CCAs) were given for defeating message blinding methods, however searching for correlation points is difficult for noise, misalignment in practical environment. In this paper, we propose an optimized cross correlation power attack for message blinding exponentiation algorithms. The attack method can select the more correlative power points of share one operation in the modular multiplication by comparing variances between correlation coefficients. Further we demonstrate that the attack method is more efficient in experiments with hardware implementation of RSA on a crypto chip card. In addition to the proposed CCA method can recovery all 1024bits secret key and recognition rate increases to 100% even when the recorded signals are noisy.

Keywords: side channel attack; correlation power analysis; cross correlation attacks; module exponentiation.

I. INTRODUCTION

Modular exponentiation algorithms are a fundamental operation in many cryptographic systems, specifically, in RSA cryptosystems. However, it has been shown that a straightforward implementation of these algorithms is vulnerable to Side Channel Attacks (SCAs) [1],such as Simple Power Analysis (SPA) [2-4], Differential Power Analysis (DPA) [5-6], Correlation Power Analysis (CPA)[7-8].

To prevent from SCAs, some countermeasures have been proposed[9-12]. DPA and SPA attacks can be prevented with message blinding methods. Message blinding works is by multiplying the message with a random, and after exponentiation removing the effect of the random value. However, the weakness of these countermeasures is that one operand in the multiplication is fixed and the same pre-computed value is used when the same bits of the private key are manipulated. Various countermeasures using the message blinding method may still be vulnerable to Cross Correlation Attacks[13-18].

Messerges, *et al.*[13] show cross-correlation signal would lead to a combined power analysis and timing but these information is not useful to differentiate between squares and multiplies. Their techniques can be modified for RSA multiply always and message blinding countermeasures by Witteman, *et al.* [14]. They used summing up correlation coefficients gathered from each multiplication power curves. An alternate approach is taken

In this paper, we propose an optimized cross correlation power attack for message blinding exponentiation algorithms. The attack method can select the more correlative power points of share one operation in the modular multiplication by comparing variances between correlation coefficients.

in [15] , where techniques to strengthen cross correlation attack by averaging correlation values and comparing to a threshold were described. The CCAs against Montgomery ladder implementation of the RSA algorithm were analyzed in [16-18], and the voting mechanism and novel one or multi reference bits approaches were proposed.

Cross Correlation Attacks (CCAs) described by these papers are proven to be effective in theory. However, in practical environment, such as noise, random time delay and random clock are applied, in their case, cross correlation values could not be used to make this distinction. This paper proposes an improved cross correlation power attack against message blinding exponentiation algorithms. This improved attack can distinguish the multiplication from not using summing up all correlation coefficients, but summing up some correlation coefficients which are selected according to bigger variance values. Experiments show the proposed method is optimized cross correlation power attack by comparing to the other three cross correlation power attacks, and the recognition rate of the private key is up to 100% with using a small number of power traces,.

The rest of this paper is organized as follows. Section II briefly reviews three of the most common modular exponentiation algorithms and describes a countermeasure algorithm using the message blinding method. Section III presents optimized attack method. Experimental results are reported in Section IV. Finally, Section V concludes the paper.

Algorithm 1 Left-to-Right binary modular exponentiation algorithm

```
Input  $x,n,d=(d_{e-1}, d_{e-2}, ...d_2, d_1, d_0)$
Output  $z=x^d \bmod n$
1. $z:=1$;
2 for $i=e-1$ down to 0
  2.1 $z=z*z \bmod n$;  --squaring
  2.2 if $d_i=1$ then
  2.3 $z=z*x \bmod n$;  -- multiplication
  2.4 end if
3 end for
4 return $z$
```

Algorithm 2 Right-to-Left binary modular exponentiation algorithm

```
Input  $x,n,d=(d_{e-1}, d_{e-2}, ...d_2, d_1, d_0)$
Output  $z=x^d$ mod n
1. $z_0:=1;z_1=x$;
2 for $i=0$ to $e-1$
  2.1 if $d_i=1$ then
  2.2 $z_0=z_0*z_1 \bmod n$;  -- multiplication
  2.3 end if
  2.4 $z_1=z_1*z_1 \bmod n$;  --squaring
3 end for
4 return $z_0$
```

II. Preliminary and Related Attacks

2.1 The binary modular exponentiation algorithms

Modular exponentiation is one of the most important arithmetic operations for public-key cryptography, such as the RSA algorithm and the ECC algorithm. The modular exponentiation is implemented by using commonly known as the "square-and-multiply" algorithm. There are two versions of efficient binary algorithm which are given in Algorithm 1 and Algorithm 2.

The Algorithm 1 is left-to-right binary method which starts at the exponent's MSB(-Most Significant Bit) and works downward, and Algorithm 2 is the right-to-left binary method, which starts at the exponent's LSB(Low Significant Bit) and works upward Here, the index e shows the bit length of the secret keys. Algorithm 1 and Algorithm 2 show these algorithms for computing z over a message x with

private exponent d.

Algorithm 3 illustrates an m-ary method of implementing modular exponentiation commonly besides Algorithm 1 and Algorithm 2. In Algorithm 3,

each iteration cycle processes m bits of the exponent. The powers $g_i(\bmod n)$ $(i=1,2,...,2^{m-1})$ can be pre-computed and used in the multiplication. The intermediate value z is raised to the power of 2^m by repeating the square operation m times.

Algorithm 1 and algorithm 2 are subject to SPA attacks[2-4] and DPA attacks[5-6]. The particular input data "n-1" SPA methods proposed by Yen et al. and "doubling attack" by proposed by Fouque et al. are effective for Algorithm 1 and algorithm 2, but are not effective for Algorithm 3. In Ref.[4], Homma et al. have presented comparative power analysis attacks based on chosen-message pairs, which can be applied to three algorithms of modular exponentiation. Hiding the relationship between the power consumption and the internal state can defeat against DPA and SPA attacks. The exponent blinding method and the message blinding method can also be regarded as effective countermeasures. We will talk the message blinding countermeasures for modular exponentiation algorithms in following section.

Algorithm 3 m-ARY modular exponentiation algorithm

```
Input  $x,n,d=(d_{e-1}, d_{e-2}, ...d_2, d_1, d_0)b,$
       where $b=2^m$ for $m\geq 1$
Output  $z=x^d$ mod $n$
 1. $g_0:=1$;
 2 for $i=1$ to $2^{m-1}$
   $g_i:= g_{i-1}*x$;
   end for
 3 for $i=t-1$ down to 0
   3.1 for $j=1$ to $m$
      $z=z*z$ mod $n$    --squaring
      end for
   3.2 $z=z*g_{di}$ mod $n$;  -- multiplication
   End for
 4 return $z$
```

Algorithm 4 SPA-DPA resistant binary modular exponentiation algorithm of the blinding message

```
Input  $x,n,d=(d_{e-1}, d_{e-2}, ...d_2, d_1, d_0),a=r-1$ mod $n$
       where $r$ is a random number
Output  $z=x^d$ mod $n$
 1. $T[0]=a$; $T[1]=x*r$ mod $n$
 2. $z=r$ mod $x$;
 3. for $i=e-1$ down to 0
   3.1 $z=z*z$ mod $n$      --squaring
   3.2 $z=z*T[d_i]$ mod $n$;  -- multiplication
   End for
 4. $z=z*a$ mod $n$;
 5. return $z$
```

2.2 Message blinding method

The message blinding method is made of randomizing message and exponent for modular exponentiation algorithms[9-12]. JaeCheol Ha, *et al.* gave the Algorithm 4 used a random number r to blind the message x.

Algorithm4 can resist Timing attacks or SPA attacks attempting to recover the private key by identifying the different modular operations, because squares and multiplications are always executed in turn. However, when an attacker could distinguish the square and multiply operations, this would not lead to attack successfully.

2.3 Cross correlation attacks on message blinding exponentiation algorithm

The typical CCAs are useful for the message blinding countermeasures[13-18]. The basic idea of CCA algorithms are that the operations would be trivially independent if two operations have no operand in common. However, if two operations share one operand there is a relation, data dependent leakage of multipliers has been shown. For example, two multiplications $a\times b$ and $c\times d$ were performed. When $a=b$ and $c\neq d$, theoretically, the correlation value of two power consumption curves is 1/2 according the hamming weights model, and when $a\neq b$, a correlation value is 0 for operands [14].

Although algorithm4 was designed to be secure against SPA and DPA attacks. But in step 3.2 of algorithm4, $z=z*T[d_i]$ mod n is calculated with the fixed pre-computed values of T[0] and T[1]. One operand of the modular multiplication is T[0] when a bit of private exponent is '0', and one operand is T[1] when a bit is '1' as shown in Figure 1.

Theoretically, each power consumptions of modular multiplication with share one op-

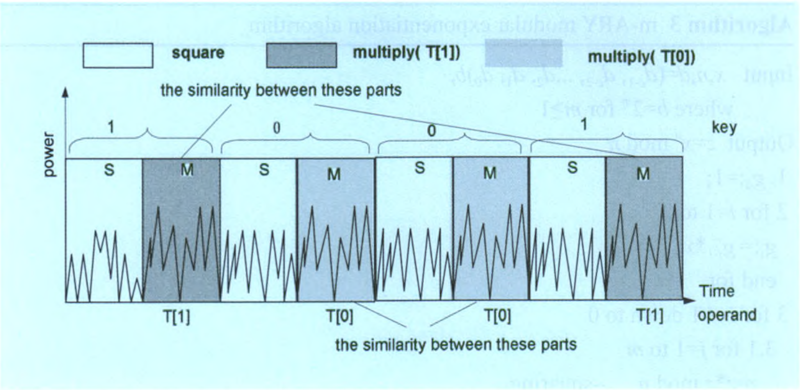


Fig.1 *Power trace of algorithm 4*

erand T[1] are higher correlation, likewise there are higher correlation between power consumptions of multiplication with share one operand T[0]. So CCAs could distinguish the operand T[0] and the operand T[1] through correlation coefficients of the power consumptions in modular multiplication. One group is composed of relatively high correlation coefficients, and the other group consists of relatively low correlation coefficients.

However, in real attack environment, the existence of noise in the power traces is inevitable. The total power consumption of the cryptosystem may then be determined as follows:

$$P_{total}=P_{op}+P_{data}+P_{el.noise}+P_{const} \quad (1)$$

Where P_{total} is the total power, P_{op} is the operation dependent power consumption, P_{data} is the data dependent power consumption, $P_{el.noise}$ denotes power resulting from the electronic noise in the hardware,

P_{const} is some constant power consumption, depending on the technical implementation in Ref. [16].

At each power point in time of every module multiplication power traces may thus be modeled by (1). Every multiplication has the same operations, so P_{op} of each point is same, P_{data} depends on two operand of the multiplication. The correlation coefficients values are mainly determined by P_{data} and $P_{el.noise}$. In practical environment, due to noise, random time delay and random clock, correlation coefficient values are almost very small and same. And probably correlation coefficient values of share one operand is lower than these of not share one operand. So, in their case, cross correlation values could not be used to make this distinction between the T[1] operand and the T[0] operand in the multiplication.

For reducing noise influence, the correlation coefficients are summed up in Ref.[14]. The averaged cross correlation coefficients of power traces are used in Ref.[15]. Akalp, *et al.* proposed improved methods summing up correlation coefficients with the bigger correlation values comparing to a threshold[16-18].

III. An Optimized Cross Correlation Attacks on Message Blinding Exponentiation Algorithm

Now, the proposed new cross correlation attacks on message blinding exponentiation algorithm is an improved method[16-18]. The basic idea of the new CCAs algorithm is that the attackers can find the power points which are less affected by noise and misalignment. In signal processing, SNR is signal and noise variance ratio. So signal has a larger variance, and noise is smaller variance. We can see correlation coefficients of power traces in modular multiplication as one signal. If the variance value of correlation coefficients in some power point is a bigger value, the noise is more less. So according to the variance values of correlation coefficients, the power points with much noise are discarded.

3.1 Preprocessing of power traces

Attacker inputs the message *x* and the secret key *d* with 1024bits into cryptosystem, then executes the Algorithm 4 of *r* times. The *r* power traces are collected with the same message and the secret key.

Firstly, Let us construct a new power trace while ignoring squaring, which can be extracted and concatenated the signals of multiplication related to step 3.2 from power traces produced during the main exponentiation of algorithm 4, as shown Figure 2.

The r new power traces can defined as T_i, $0 \le i < r$. In addition, we define a new power trace $T_i = M_{i,0}||M_{i,1}|| \ldots ||M_{i,1022}||M_{i,1023}$. We get r new power traces to construct the matrix X below.

$$X = \begin{bmatrix} M_{0,0} & M_{0,1} & \cdots & M_{0,1022} & M_{0,1023} \\ M_{1,0} & M_{1,1} & \cdots & M_{1,1022} & M_{1,1023} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ M_{r-2,0} & M_{r-2,1} & \cdots & M_{r-2,1022} & M_{r-2,1023} \\ M_{r-1,0} & M_{r-1,1} & \cdots & M_{r-1,1022} & M_{r-1,1023} \end{bmatrix} \tag{2}$$

Each row vector represents a power trace with only multiplications. $M_{i,j}$ represents a multiplication including L points as follows:

$$M_{i,j} = [p_{i,j*L}, p_{i,j*L+1}, \cdots, p_{i,j*L+L-1}] \tag{3}$$

Where $0 \le i < r$, $0 \le j < 1024$. Each column vector in matrix X is defined as follows:

$$x_{j*L+k} = [p_{0,j*L+k}, p_{1,j*L+k}, \cdots, p_{r-2,j*L+k}, p_{r-1,j*L+k}]^T \tag{4}$$

where $0 \le k < L$, $0 \le j < 1024$.

In Figure 2, between the vector u_1 and the vector u_2 the Person correlation[14] can be computed as follows:

$$\rho(x_{u_1}, x_{u_2}) = \frac{\sum_{i=0}^{r-1}(p_{i,u_1} p_{i,u_2}) - \frac{\sum_{i=0}^{r-1} p_{i,u_1} \sum_{i=0}^{r-1} p_{i,u_2}}{r}}{\sqrt{\left(\sum_{i=0}^{r-1}(p^2_{i,u_1}) - \frac{\left(\sum_{i=0}^{r-1} p_{i,u_1}\right)^2}{r}\right)\left(\sum_{i=0}^{r-1}(p^2_{i,u_2}) - \frac{\left(\sum_{i=0}^{r-1} p_{i,u_2}\right)^2}{r}\right)}} \tag{5}$$

3.2 Optimized cross correlation attacks process

The proposed new attack process is given as follows.

Input: the matrix X

Output: Secret Key d.

Step 1. Calculate the Pearson correlation estimate of the matrix X according the formula 5. We can get correlation coefficients matrix:

$$\widetilde{CT} = [CT_0, CT_1 \cdots, CT_{1023}]^T$$

Where $CT_0, CT_1, \ldots, CT_{1022}, CT_{1023}$ are correlation values between the other multiplication and the first multiplication.

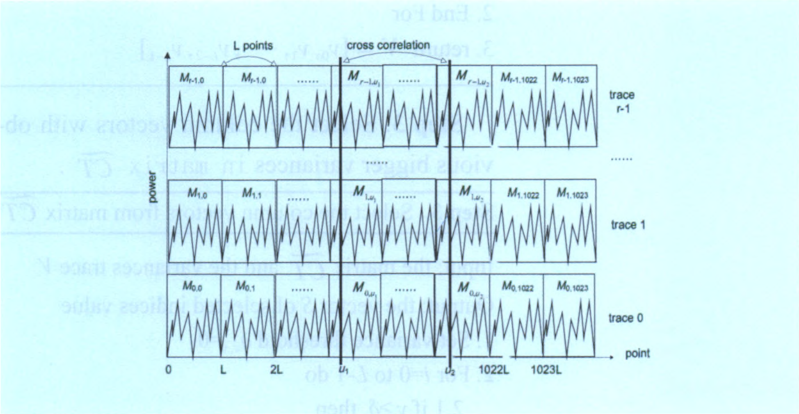


Fig.2 *r power traces of the multiplication*

$$\widetilde{CT} = \begin{bmatrix} \xi_{0,0} & \xi_{0,1} & \cdots & \xi_{0,L-2} & \xi_{0,L-1} \\ \xi_{1,0} & \xi_{1,1} & \cdots & \xi_{1,L-2} & \xi_{1,L-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \xi_{1022,0} & \xi_{1022,1} & \cdots & \xi_{1022,L-2} & \xi_{1022,L-1} \\ \xi_{1023,0} & \xi_{1023,1} & \cdots & \xi_{1023,L-2} & \xi_{1023,L-1} \end{bmatrix} \tag{6}$$

Step 1. Calculate Pearson correlation value of X

```
Input: the matrix X of power traces
Output: the matrix $\widetilde{CT}$ of correlation coefficients
1 For $j$=0 to 1023 do
    1.2 For $i$=0 to $L$-1 do
        1.2.1 Set the column vector
            $x_i = [p_{0,i}, p_{1,i}, \cdots, p_{r-2,i}, p_{r-1,i}]^T$
        1.2.2 Set the column vector
            $x_{j*L+i} = [p_{0,j*L+i}, p_{1,j*L+i}, \cdots, p_{r-2,j*L+i}, p_{r-1,j*L+i}]^T$
        1.2.3 Calculate correlation values
            between and $x_{j*L+i}$
            $\xi_{j,i} = \rho(x_i, x_{j*L+i})$
    1.3 End For
2. End For
3. return $\widetilde{CT} = [CT_0, CT_1, \cdots, CT_{1022}, CT_{1023}]^T$
```

Step 2. Calculate variance of every column vector in matrix $\widetilde{CT}$. $V = [v_0, v_1, \ldots, v_{L-2}, v_{L-1}]$ is the variance trace and D represents calculating variance.

Step 2. Calculate variance in matrix $\widetilde{CT}$

```
Input: The matrix $\widetilde{CT}$
Output: The vector $V$ of variances
1 For $i$=0 to $L$-1 do
    1.1 Set the column vector
        $y_i = [\xi_{0,i}, \xi_{1,i}, \cdots, \xi_{1022,i}, \xi_{1023,i}]^T$
    1.2 Calculate variances of the vector $y_i$
        $v_i = D(y_i)$
```

```
2. End For
3. return $V = [v_0, v_1, \cdots, v_{L-2}, v_{L-1}]$
```

Step 3. Select the column vectors with obvious bigger variances in matrix $\widetilde{CT}$.

Step 3. Select the column vectors from matrix $\widetilde{CT}$

```
Input: the matrix $\widetilde{CT}$ and the variances trace $V$
Output: the vector $S$ of selected indices value
1. Set variance threshold $\delta_{v,\mathrm{t}}$=0
2. For $i$=0 to $L$-1 do
   2.1 if $v_i \geq \delta_v$ then
      2.1.1 $\mu_j = i$ ;
      2.1.2 $t = t+1$
   2.2 End if
3. End For
4. return the indices $S = [u_0, u_1, \dots, u_{t-2}, u_{t-1}]$
```

The threshold with bigger variances δ_v is decided by directly observing the graph of variances vector V and finding the threshold with significantly higher variances. S is the indices of bigger than δ_v in variances vector V.

Step4. Sum up correlation coefficient of row vector according indices S in matrix $\widetilde{CT}$.

Step 4. Sum up seleted correlation coefficient

```
Input: matrix $\widetilde{CT}$ and the indices $S$.
Output: the vector $C$ of summation
1. For $i$=0 to 1023 do
 1.1 Set $c_i$=0
 1.2 For $j$=0 to $t$-1 do
    1.2.1 $c_i = c_i + \xi_{i,\mu_j}$
 1.3 End For
2.End For
3. Return the vector $C = [c_0, c_1, \dots, c_{1023}]$
```

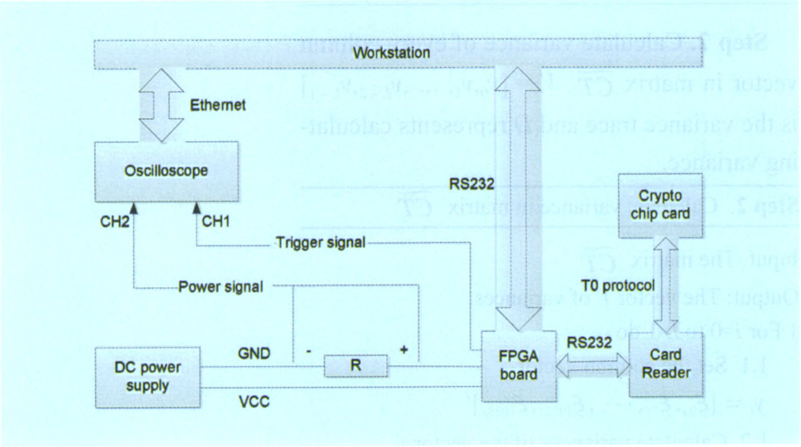


Fig.3 *Components for Power analysis attack*

the vector C is composed of selected correlation coefficient summation for one multiplication.

Step 5. Finally, Conclude the secret key by dividing the vector C into two group.

Step 5. Conclude the Secret Key d

```
Input : the vector C, the threshold $\delta_c$
Output: the secret key $d$
1. For $i$=0 to 1023 do
   1.1 if $c_i \geq \delta_c$ then
         $d_{1023-i} = 1$
       else
         $d_{1023-i} = 0$
   1.2 End If
2 End For
3. return $d = (d_{e-1}, d_{e-2}, \dots d_2, d_1, d_0)$
```

The threshold δ_c can simply be set by averaging the second bigger value in the vector C. According the threshold δ_c, we can distinguish between the T[1] operand and the T[0] operand in the multiplication, and decide the secret key.

IV. EXPERIMENTS RESULT

4.1 Experimental Environments

In this section attacker tests CCAs against Algorithm 4 in the smartcard with Montgomery multiplier. The components for power analysis attacks is shown in Figure 3. These components usually interact with each other[19-20]. The workstation sends commands to the FPGA board. The FPGA board receives commands and forwards the commands to the Card Reader that triggers the execution of Algorithm 4 in the crypto chip card. During the execution of Algorithm 4, the oscilloscope can collect the voltage signal(power trace) from the two end of resistance connected the FPGA board. The workstation receives the recorded power trace from the oscilloscope via an Ethernet interface.

4.2 Experimental results for the previous CCAs

In this section attacker tests three CCAs method [13-18] for Algorithm 4 in the power analysis platform. Attacker inputs the plaintext x and the secret key d with 1024bits into the crypto chip card. Algorithm 4 runs and the oscilloscope collects the power traces.

The Figure 4(a) shows the power trace of Algorithm 4 with 1024 bits of the exponent(secret key) and the plaintext x. To simplify the attack description, we only give attack process of the most significant 32bits of the secret key in detail, and other bits of the secret key too processing. The most significant 32 bits of the exponent is 0xCA0A4410(Hex) in the Figure 4(b).

In Figure 4(b), square and multiply alternate operations according Algorithm 4. The algorithm 4 is repeated running r times with the same plaintext and the exponent.

Firstly, We can reconstruct new multiplication power traces T_0, T_1, ...,T_{r-2}, T_{r-1} by extracting r power traces and concatenating the signals of multiplication.

Secondly, after the r power traces can reconstruct a block matrix X, we can calculate the Pearson correlation according the formula 5 and get correlation traces CT_0, CT_1, ..., CT_{1022}, CT_{1023} as shown in Figure 5. CT_i is similar correlation distribution. We can not classify the same bits and the different bits compared from correlation coefficients directly.

According the CCA method in Ref.[14], we sum up all the correlation coefficients corresponding to each multiplication signal as shown in Figure 6(a).

In Figure 6(b), If a higher value than the threshold, this bit and the first bit have the same value, i.e. an exponent bit is '1' (red mark). Otherwise, this bit is different from the first bit, namely the bit is '0'. If observers set the specific threshold 2.2, the most significant 32bits of the exponent are concluded 1110 1111 1010 1111 0110 0111 0111 0011(0xEFAF6773). The concluded result has 14 error bits. If the threshold 2.8, the most significant 32bits of the exponent are decided 1110 1110 0010 1110 0100 0100 0001 0011 (0xEE2E4413) .The error bits reduce to 6 bits.

We average correlation coefficients summa-

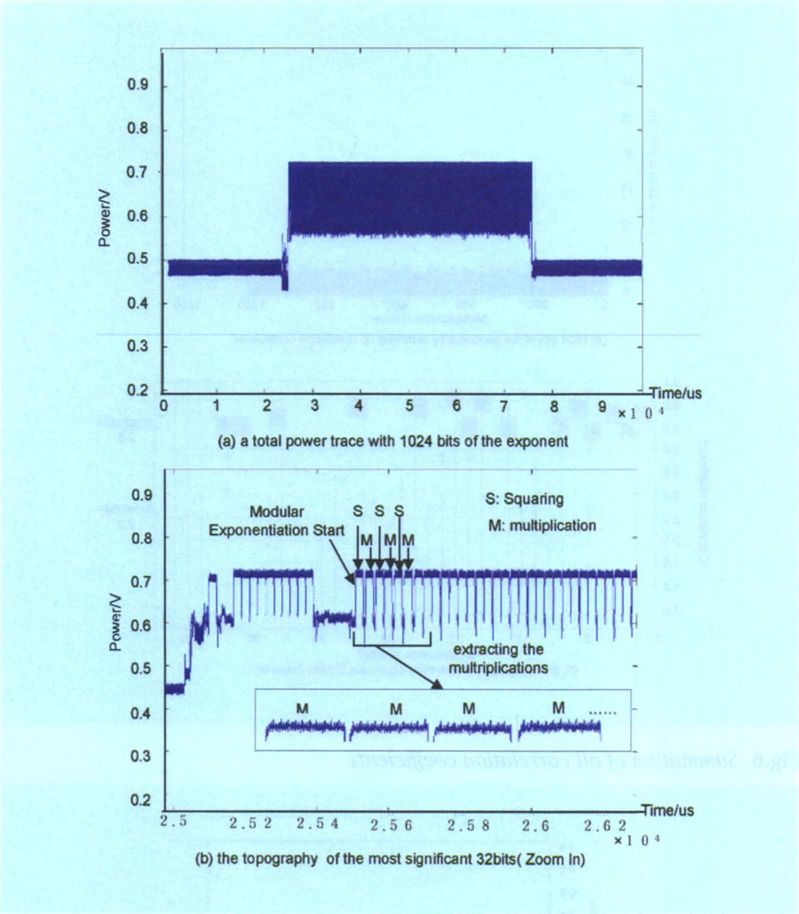


Fig.4 *Power trace of algorithm 4 by low pass filter*

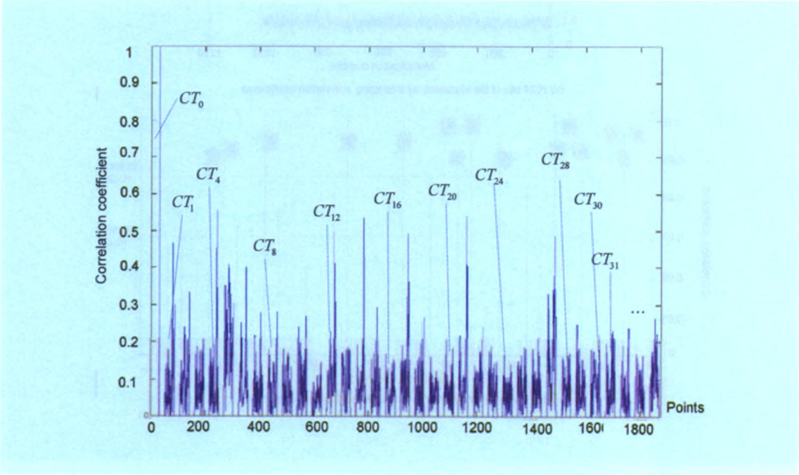
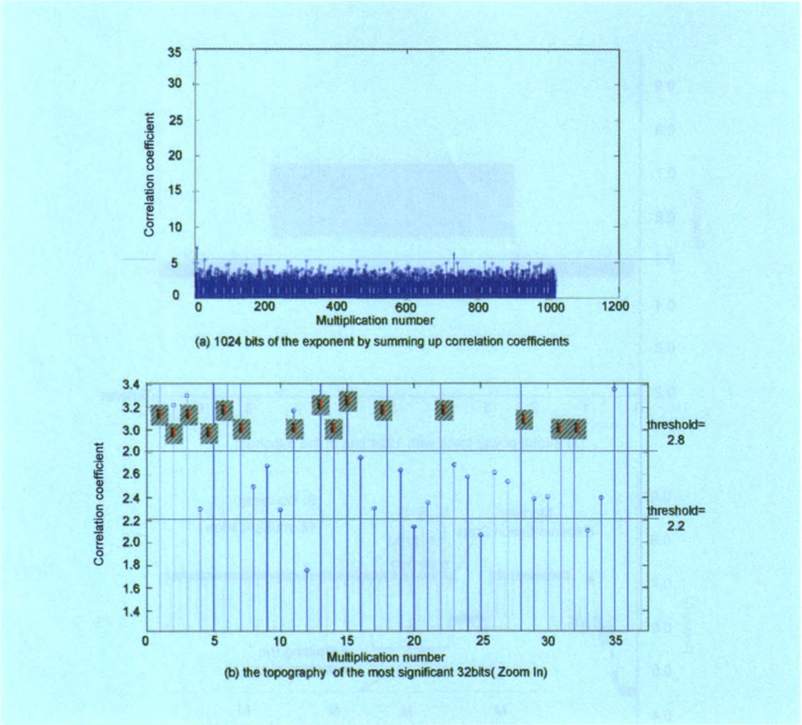


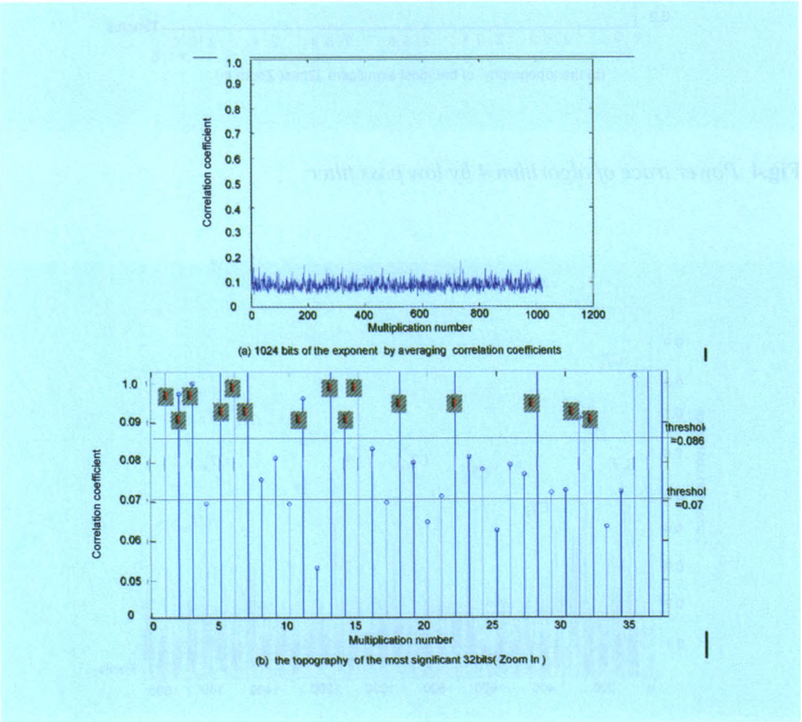
Fig.5 *correlation traces using 100 power traces*



(a) 1024 bits of the exponent by summing up correlation coefficients

(b) the topography of the most significant 32bits(Zoom In)

Fig.6 *Summation of all correlation coefficients*



(a) 1024 bits of the exponent by averaging correlation coefficients

(b) the topography of the most significant 32bits(Zoom In)

Fig.7 *Correlation coefficients average values*

tion of each multiplication[15]. The attack result of the most significant 32bits is shown as Figure7(b). The result is 0xEFCF6F7F if the threshold is 0.07 ,there is 16 error bits. if the threshold is 0.0086, the result is 0xEE2E4413 with 6 error bits.

According Ref.[16-18], we sum up the correlation coefficients with bigger than the threshold, but not all correlation coefficients the corresponding to each multiplication.

If the threshold is 0.05, the summation of correlation coefficients greater than 0.05 is shown Figure 8(a). Then the decided 32bits is 0xCBAF6111 with 7 error bits if the threshold is 2.5, And if the threshold is 3.0, the concluded 32bits is 0xCB096110 with 4 error bits.

Above three CCA method, the selected threshold affects the accuracy of the attack. It is difficult to select the optimal threshold.

4.3 Experimental results for the proposed CCAs

In this section, attackers use the improved CCA. According Step 2 of the new CCA, The variance vector values of every column vector in matrix $\widetilde{CT}$ are calculated. The variance trace V is as shown Figure 9.

In Figure 9, observers can know that 12 to 14 and 17 to 18 points have the highest variance values. We summed up correlation coefficients of these location power points. Figure 10(a) shows the total result of 1024 bits.

Attacker can easily get second largest correlation value is about 0.82. So the specific threshold is about 0.41 by averaging the second largest value as shown in Figure 10(b). Then the most significant 32bits of the exponent is 0xCA0A4410 by Step 5. The most significant 32bits is identical to the input the exponent 0xCA0A4410. In this case, we were able to break the 1024bit exponent successfully.

4.4 The contrast of the results

The analysis in the former section shows that the improved CCA is an effective attack method. TABLE 1 list the attack result accuracy of the four attack methods. The results came from the different sampling rate and the different amount of power traces.

The accuracy of from the Table 1 shows that the new proposed attack results is the best and the private key recognition rate increases to 100% in 300 power traces. And bigger sampling rate and more power traces can get a high accuracy ,but increasing the number of power traces is greater than increasing the sampling rate.

V. Conclusions

This paper proposes a improved Cross Correlation Attacks against message blinding exponentiation algorithms in real attack environment. The more correlative power points of share one operation in the modular multiplication are selected by correlation coefficient variance threshold. It can identify the modular multiplication automatically and needn't the knowledge about plaintext and ciphertext. The experimental result shows that the proposed method is optimized CCA by comparing to the other three CCAs. This algorithm improved the secret key recognition rate up to 100% in 300 power traces. In the paper, the proposed CCA attacks only a smart card. This new potential work should then be taken into account when changing cryptographic devices, more power traces are required.

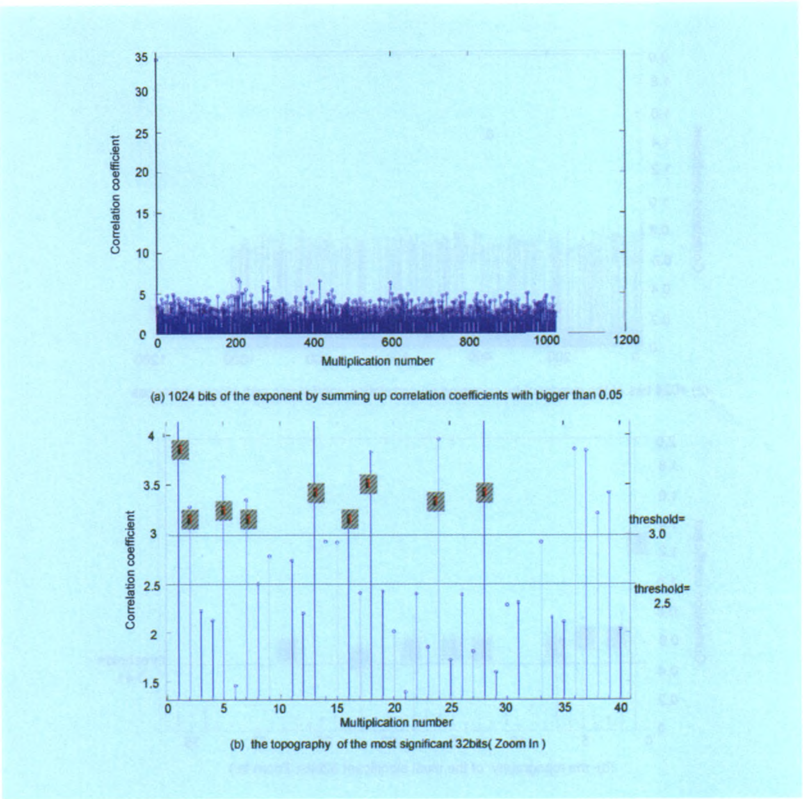


Fig.8 *Summation of selected correlation coefficients*

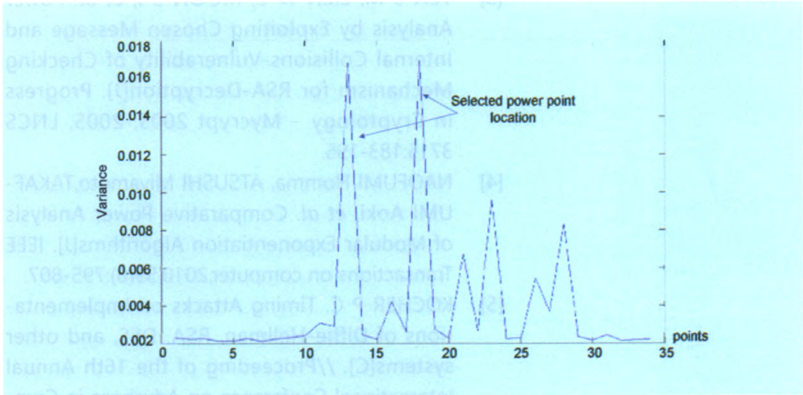


Fig.9 *Variance of correlation coefficients*

Acknowledgements

The authors would like to thank the reviewers for their detailed reviews and constructive comments, which have helped improve the quality of this paper. This work was supported in part by National Natural Science Foundation of China Project (Grant No. 60873216); Scientific and Technological Research Priority Projects of Sichuan Province(Grant No. 2012GZ0017).

Table I *Contrast of result*

Sampling rate	Amount of traces	Kim CCA	Witteman CCA	Akalp CCA	New CCA
1M	500	80.4%	75.6%	88.4%	100%
2.5M	500	82.2%	76.3%	92.3%	100%
1M	300	74.4%	70.3%	85.3%	100%
2.5M	300	75.8%	73.2%	86.7%	100%
1M	100	69.3%	60.5%	78.1%	99.9%
2.5M	100	70.5%	61.2.%	79.5%	99.9%

References

[1] KOCHER P C, JAFFE J, JUN B. Differential Power Analysis[C].//Proceeding of the 19th Annual International Conference on Advances in Cryptology(CRYPTO'99):August 15-18,1999,Santa Barbara, California, USA: Springer, 1999: 388-397.

[2] FOUQUE A P, VALETTE F. The Doubling Attack—Why Upwards is Better Than Downwards[J]. Cryptographic Hardware and Embedded Systems (CHES '03),2003: 269-280.

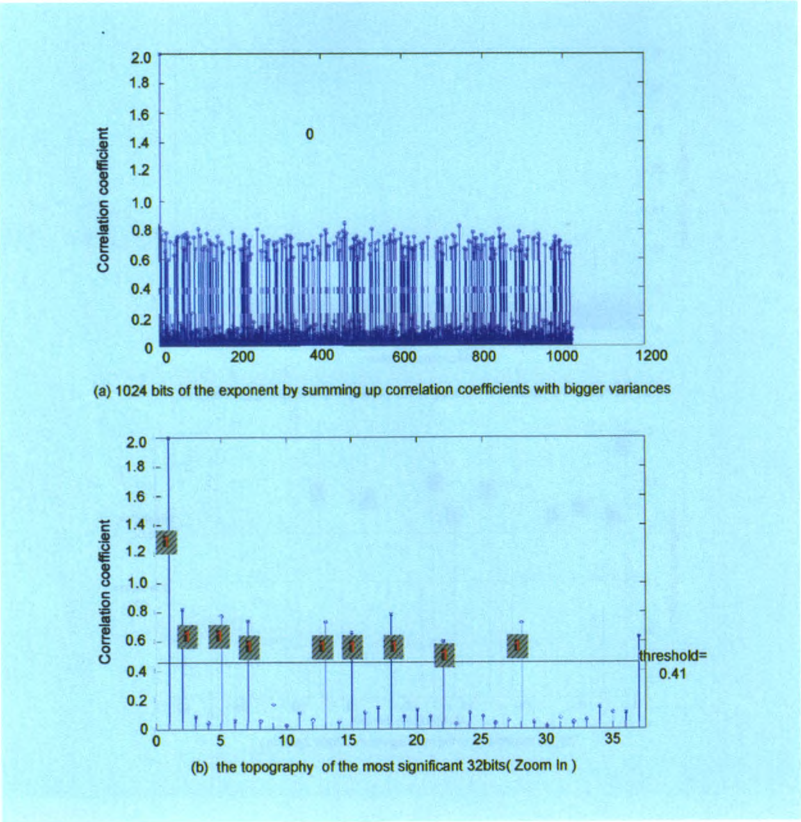


Fig.10 *summation of correlation coefficients with the higher variance values*

[3] YEN S M, LIEN W C, MOON S J, *et al.* Power Analysis by Exploiting Chosen Message and Internal Collisions-Vulnerability of Checking Mechanism for RSA-Decryption[J]. Progress in Cryptology – Mycrypt 2005, 2005, LNCS 3715:183-195.

[4] NAOFUMI Homma, ATSUSHI Miyamoto,TAKAFUMI Aoki, *et al.* Comparative Power Analysis of Modular Exponentiation Algorithms[J]. IEEE Transactions on computer,2010,59(6):795-807.

[5] KOCHER P C. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and other systems[C]. //Proceeding of the 16th Annual International Conference on Advances in Cryptology(CRYPTO'96): August 18-22,1996,Santa Barbara, California, USA: Springer, 1996:104-113.

[6] MESSERGES T S, DABBISH E A, SLOAN R H. Investigations of Power Analysis Attacks on Smartcards [C]. //Proceeding of the USENIX Workshop Smartcard Technology: May 10-11,1999. Chicago, Illinois, USA: IEEE Press,1999:151-161.

[7] BRIER E, CLAVIER C, OLIVIER F. Correlation Power Analysis with A Leakage Model[C].// Proceeding of Cryptographic Hardware and Embedded Systems-CHES2004: August 11-13, 2004, Marriott Cambridge, Boston, USA: Springer, 2004,LNCS 3156:16-29.

[8] AMIEL F, FEIX B, VILLEGAS K. Power Analysis for Secret Recovering and Reverse Engineering of Public Key Algorithms[C]. //Proceeding of the 14th International Workshop on Selected Areas in Cryptography(SAC2007): August 16-17, 2007,Ottawa, Canada: Springer, 2007,LNCS 4876: 110-125.

[9] CORON J S Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems[C]. //Proceeding of Cryptographic Hardware and Embedded Systems-CHES1999: August 12-13, 1999,Worcester, Massachusetts, USA: Springer, 1999,LNCS 1717 :292-302.

[10] YEN S M, KIM S J, LIM S G, *et al.* A Countermeasure Against One Physical Cryptanalysis May Benefit Another Attack[C]. //Proceeding of 4th International Conference on Information Security and Cryptology(ICISC 2001): December 6-7, 2001,Seoul, Korea: Springer, 2002,LNCS 2288:414-427.

[11] IZU T, TAKAGI T. A Fast Parallel Elliptic Curve Multiplication Resistant against Side Channel Attacks[C]. //Proceeding of 5th International Workshop on Practice and Theory in Public Key Cryptosystems(PKC2002): February 12-14,2002 Paris, France: Springer, 2002,LNCS 2274:280-296.

[12] HA J C, JUN C H, PACK J H, *et al.* A New CRT-RST Scheme Resistant to Power Analysis and Fault Attack[C]. //Proceeding of 3th on Convergence and Hybrid Information Technology (ICCIT2008): November 11-13,2008,Busan, Korea, IEEE Computer Society Press ,2008:351-356.

[13] MESSERGES T S, DABBISH E A, SLOAN R H. Power Analysis Attacks of Modula Exponentiation in Smartcards[C]. //Proceeding of Cryptographic Hardware and Embedded Systems-CHES1999: August 12-13, 1999,Worcester, Massachusetts, USA: Springer, 1999,LNCS 1717:144-157.

[14] WITTEMAN M F, JASPER G J, VAN W, *et al.* Defeating RSA Multiply-Always and Message Blinding Countermeasures[C]. //Proceeding of the Cryptographers' Track at the RSA Conference(CT-RSA 2011):February 14-18,2011,San Francisco, CA, USA: Springer, 2011,LNCS 6558:77-88.

[15] KIM H S, KIM T H, YOON J C, *et al.* Practical Second-Order Correlation Power Analysis on the Message Blinding Method and Its Novel Countermeasure for RSA[J] ETRI Journal, 2010, 32(1):102-111.

[16] AKALP K E, SOYSAL B. SOYSAL M, *et al.* New Cross Correlation Attack Methods on the Montgomery Ladder Implementation of RSA[C].// Proceeding of the 3rd IEEE International Advance Computing Conference (IACC2013): February 22-23, 2013,Ghaziabad, India : IEEE Press,2013:138-142.

[17] AKALP K E, TANGEL A. All Bits Cross Correlation Attack on the Montgomery Ladder Implemen-

tation of RSA[C]. //Proceeding of the 18th International Conference on Digital Signal Processing (DSP2013) : July 1-3, 2013,Santorini, Greece: IEEE Press,2013:
183-187.

[18] AKALP K E, TANGEL A. A New Style CPA Attack on the ML Implementation of RSA[C]. //Proceeding of the 18th International Computer Science and Engineering Conference (ICSEC2014): July 30-August 1, 2014,Khon Kaen, Thailand: IEEE Press,2014:323-328.

[19] CAO Nana,WAN Wunan,CHEN Yun, *et al.* Chosen plaintext SPA Attacks RSA Algorithms of 8051 Chip[J].Journal of Chengdu University of Information Technology,2011,26(4):356-361.

[20] CHEN Aidong, XU Son, CHEN Yun,QIN Zhiguang. Collision based on chosen message sample power clustering attack algorithm[J]. China Communications,2013,5:114-119.

Biographies

WAN Wunan, currently received her Ph.D. degrees in Chengdu Institute of Computer Applications, Chinese Academy of Sciences, Chengdu, in 2006. Her has been an associate professor in Chengdu University of Information Technology. Her research interests include side channel attack, distribution storage security and codes theory. *The corresponding author. Email:nan_wwn@cuit.edu.cn

YANG Wei, born in 1988, is currently a master candidate of Chengdu University of Information Technology, China. His research interests is side channel attack and network security. Email: ywhpu0802@163.com.

CHEN Jun, has been an associate professor in Chengdu University of Information Technology. His research interests include side channel attack and codes theory. Email:chenjun@cuit.edu.cn