SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2019, Vol. 62 032105:1–032105:14
https://doi.org/10.1007/s11432-018-9515-9

Partially known information attack on SM2 key exchange protocol

Wei WEI[1*], Jiazhe CHEN[1], Dan LI[2] & Beibei WANG[1]

[1]*China Information Technology Security Evaluation Center, Beijing* 100085*, China;*
[2]*Institute for Advanced Study, Tsinghua University, Beijing* 100084*, China*

Received 15 May 2018/Accepted 4 July 2018/Published online 24 January 2019

Abstract SM2 key exchange protocol is a part of the SM2 public key cryptographic algorithm based on elliptic curves which has been issued by Chinese State Cryptography Administration since 2010. Under the guide of Chinese government, SM2 has been widely used in Chinese commercial applications. This paper gives the first partially known information attack on SM2 key exchange protocol. Our attack is based on a technique modified from the hidden number problem (HNP) which was introduced originally to study the bit security of Diffie-Hellman and related schemes. We present a polynomial-time algorithm which could recover the user's secret key when given about half least significant bits of the two unknown intermediate values in each congruence over about 30 to 40 instances. Compared with the standard HNP, our approach deals with congruence involved two independent unknown variables and each of them possesses the same size as the secret key. Moreover, our results almost coincide with the previous best result among the same field considering the extreme case in which one variant is completely revealed.

Keywords SM2 key exchange protocol, cryptanalysis, information leakage, lattice attack, extended hidden number problem

Citation Wei W, Chen J Z, Li D, et al. Partially known information attack on SM2 key exchange protocol. Sci China Inf Sci, 2019, 62(3): 032105, https://doi.org/10.1007/s11432-018-9515-9

1 Introduction

SM2 public key cryptographic algorithm based on elliptic curves is a national standard published by Chinese State Cryptography Administration in 2010 [1], which originally aims to guide the manufacturers on developments of information security products for commercial uses in China. Recently, SM2 has more and more extensive application in international security fields such as its approval in TPM 2.0 [2] and ISO/IEC 14888-3 [3]. SM2 mainly contains three parts which are digital signature algorithm, public key encryption algorithm, and key exchange protocol. In this paper, we mainly focus on the SM2 key exchange protocol.

SM2 key exchange protocol is a variant of elliptic curve Diffie-Hellman key exchange protocol [4]. Recall the Diffie-Hellman key exchange protocol. Let $(G, \cdot)$ be a group with $g \in G$, there are two participants who respectively choose random u and v. They compute g^u and g^v and then exchange it to each other. Then the Diffie-Hellman secret key is g^{uv}. Theoretically, the security of SM2 key exchange protocol is based on the computational intractability of the discrete logarithm problem on elliptic curves (ECDLP). ECDLP-based cryptosystems need smaller size of parameters to enjoy the same security level in contrast to many other well known public-key schemes based on hard problems over finite fields. Nevertheless,

* Corresponding author (email: weiw@itsec.gov.cn)

there still exist many practical cryptographic vulnerabilities during implementations caused by side-channel attacks, fault injection attacks or software bugs. Therefore, special cares must be taken with the nonce and other intermediate values during the establishment of shared keys. In this paper, we mainly study partially known information attack on SM2 key exchange protocol. Concretely, with some leakage of the intermediate values during the computation of the shared key on one of the participants, we aim to disclose the private key.

Many types of attacks against information leaks in ECDLP-based public-key cryptosystems such like digital signature algorithm (DSA) and elliptic curves digital signature algorithm (ECDSA) have been carried out [5–8]. However there are fewer results on the security of ECDLP-based key exchange protocol, especially for SM2. In 1996, Boneh and Venkatesan [9] first studied the nonce leakage from Diffie-Hellman and other related schemes in prime fields by utilizing a tool named the hidden number problem (HNP) [9], and they proved the bit security of the $\sqrt{\log p}$ most significant bits of the secret key. Indeed, HNP could be regarded as a multidimensional linear congruences of truncated variables. It plays important roles not only in the proof of bit security of Diffie-Hellman and related schemes [10], but also in attacks against DSA-like signature schemes. In 1999, Howgrave-Graham and Smart [5] attacked the 160 bits DSA, based on a reasonable number of signatures with knowledge of some bits from each corresponding nonce. Subsequently, Nguyen and Shparlinski [8] gave a provable polynomial-time attack under some assumptions against DSA in which the nonces are partially known. This result has been improved in [6]. A similar partially known nonces attack has been implemented to the 256 bits SM2 digital signature scheme [11], in which one could recover the private key by 100 signatures with knowledge of 3 bits of each nonce. Most of the above attacks succeeded under the fact that the security of related schemes is guaranteed by the randomness of nounces chosen in the generation step. Different with this, statistical biases derived from partially information leakage have been exploited for key recovery as well [12–15]. This type of attacks are expected to succeed with knowledge of fewer bits at the cost of larger number of instances.

HNP usually deals with issues related to one unknown target together with a number of its random approximations. A real-world cryptanalytic problem arises when the given approximations are inconsecutive, which means the disclosed bits distribute discretely. To relax the restriction on uniformity and improve the usability of HNP for practical attacks, various variants of this problem adopted in different applications have appeared [8, 16, 17]. In 2006, Hlaváč and Rosa [18] presented an extended hidden number problem (EHNP) and a polynomial time algorithm for recovering the secret key. EHNP significantly relaxes the strict limitations on the position of information leakage in HNP and this is very important to current side channel attacks. The technique was then improved and applied to attack OpenSSL implementions with windowed non-adjacent form (wNAF) on ECDSA [19].

Lattice basis reduction algorithm is a crucial tool in solving HNP (and EHNP), because a short lattice vector or a close lattice point of the related lattice often reveals the solution to HNP (or EHNP). Lattice basis reduction has been a problem of lasting interest in lattice-based cryptographic constructions and cryptanalysis. Since the seminal work of the LLL algorithm [20] proposed in 1982, many improved algorithms have appeared subsequently. The most practical and widely-used algorithm by now is BKZ 2.0 proposed by Chen and Nguyen in 2011 [21], which is improved from the Schnorr-Euchner's BKZ [22]. In recent years, some variants of BKZ which perform better in practice have been investigated [23–25].

Up to now, few results about partially information attack against elliptic curve key exchange protocol could be found in the literature. Compared with other DSA-like schemes on the side of partially information attack, the case of SM2 key exchange protocol is more intractable because there exist additional unknown random variables related to the nonces, and this makes the private key still be covered, even if the nonces can be completely disclosed. In this paper, we deal with issues when some partially information about the intermediate variables are leaked in the generation of SM2 shared key, which is much more natural to achieve in realistic scenario. This is the first result about partially information leak attack against SM2 key exchange protocol. To recover the private key from a set of instances, we first transform it to an EHNP-like problem and then use lattice basis reduction as a crucial tool to reduce the problem to a closest vector search problem in related lattice corresponding to some target vector. Though the

construction is very similar to classical technique, the proof of success probability is more intricate due to the subtle difference between the problem and the standard EHNP. Furthermore, our experiments show that, given about 40 instances with 512 unknown bits in each one, if several more than 265 least or most significant bits of the intermediate variables are known, we can recover the maximum corresponding part of private key in a personal laptop with success probability larger than 0.95. Moreover, the known number of bits can decrease to 259 under proportional optimization. This result almost coincides with the previous best result [11] among the same field when considering the extreme case in which one variant is completely revealed.

This paper is organized as follows: Section 2 is the preliminaries where the introduction of SM2 key exchange protocol and some necessary background are included. In Section 3, we give our attack on SM2 key exchange protocol with some partially information known. Section 4 shows our experimental results on the attack proposed in Section 3. Section 5 concludes the paper.

2 Preliminaries

In this section, we briefly review the SM2 key exchange protocol, together with some basic knowledge of lattice. The notations of HNP and EHNP are also introduced as crucial tools to our attack.

2.1 Overview of SM2 key exchange protocol

In the study of SM2 key exchange protocol within this paper, we focus on elliptic curves defined on prime fields $\mathbb{F}_p$ with characteristic $p > 3$. For parameters $a, b \in \mathbb{F}_p$ satisfying $4a^3 + 27b^2 \neq 0$, the group $\mathbb{E}(\mathbb{F}_p)$ formed by rational points of the elliptic curve and the infinity point $\mathcal{O}$ is defined as

$$\mathbb{E}(\mathbb{F}_p) = \{P = (x, y) \mid y^2 = x^3 + ax + b \bmod p,\ x, y \in \mathbb{F}_p\} \cup \{\mathcal{O}\}.$$

In the key generation period, choose a base point $G = (x_G, y_G) \in \mathbb{E}(\mathbb{F}_p)$ with prime order n. The SM2 protocol client A (and B, respectively), randomly selects $d_{\mathrm{A}} \in [1, n-1]$ (and $d_{\mathrm{B}} \in [1, n-1]$, respectively) as its private key. The corresponding public key is $P_{\mathrm{A}} = d_{\mathrm{A}}G$ ($P_{\mathrm{B}} = d_{\mathrm{B}}G$, respectively). Moreover, some other necessary notations used in SM2 key exchange protocol are listed as follows. One can refer to the official standard [1] for integrated description.

(1) h: cofactor, $h = \#\mathbb{E}(\mathbb{F}_p)/n$, where $\#\mathbb{E}(\mathbb{F}_p)$ is the size of $\mathbb{E}(\mathbb{F}_p)$.
(2) klen: the bit length of the session key.
(3) KDF(Z, klen): one-way derivation hash function with output length klen.
(4) Z_{A}: hash value of client A's identification.
(5) Z_{B}: hash value of client B's identification.
(6) Hash(): hash function.
(7) $\|$: concatenation of two strings.

The main procedure of SM2 key exchange protocol is as follows. As the initiator, client A will establish a session key with client B. Denote $w = \lceil(\lceil\log_2(n)\rceil/2)\rceil - 1$.

Client A.

(1) Randomly choose an integer $r_{\mathrm{A}} \in [1, n-1]$.
(2) Compute $R_{\mathrm{A}} = r_{\mathrm{A}}G = (x_1, y_1)$.
(3) Send R_{A} to client B.

Client B.

(1) Randomly choose an integer $r_{\mathrm{B}} \in [1, n-1]$.
(2) Compute $R_{\mathrm{B}} = r_{\mathrm{B}}G = (x_2, y_2)$, $\bar{x}_2 = 2^w + (x_2 \& (2^w - 1))$, $t_{\mathrm{B}} = (d_{\mathrm{B}} + \overline{x}_2 \cdot r_{\mathrm{B}}) \bmod n$, $\bar{x}_1 = 2^w + (x_1 \& (2^w - 1))$ from R_{A}, $V = (h \cdot t_{\mathrm{B}})(P_{\mathrm{A}} + \bar{x}_1 R_{\mathrm{A}}) = (x_V, y_V)$, and $K_{\mathrm{B}} = \mathrm{KDF}(x_V \,\|\, y_V \,\|\, Z_{\mathrm{A}} \,\|\, Z_{\mathrm{B}}, \mathrm{klen})$.
(3) (Optional for key confirmation) Compute $S_{\mathrm{B}} = \mathrm{Hash}(0x02 \,\|\, y_V \,\|\, \mathrm{Hash}(x_V \,\|\, Z_{\mathrm{A}} \,\|\, Z_{\mathrm{B}} \,\|\, x_1 \,\|\, y_1 \,\|\, x_2 \,\|\, y_2))$.
(4) Send R_{B} (and S_{B}, optional for key confirmation) to client A.

Client A.

(1) Compute $\bar{x}_1 = 2^w + (x_1 \& (2^w - 1))$, $t_{\rm A} = (d_{\rm A} + \overline{x}_1 \cdot r_{\rm A}) \bmod n$, $\bar{x}_2 = 2^w + (x_2 \& (2^w - 1))$ from $R_{\rm B}$, $U = (h \cdot t_{\rm A})(P_{\rm B} + \bar{x}_2 R_{\rm B}) = (x_U, y_U)$, and $K_{\rm A} = {\rm KDF}(x_U \parallel y_U \parallel Z_{\rm A} \parallel Z_{\rm B}, {\rm klen})$.

(2) (Optional for key confirmation) Compute $S_1 = {\rm Hash}(0x02 \parallel y_U \parallel {\rm Hash}(x_U \parallel Z_{\rm A} \parallel Z_{\rm B} \parallel x_1 \parallel y_1 \parallel x_2 \parallel y_2))$ and verify if $S_1 = S_{\rm B}$ holds. Terminate if it is not true.

(3) (Optional for key confirmation) Compute $S_{\rm A} = {\rm Hash}(0x03 \parallel y_U \parallel {\rm Hash}(x_U \parallel Z_{\rm A} \parallel Z_{\rm B} \parallel x_1 \parallel y_1 \parallel x_2 \parallel y_2))$, and send $S_{\rm A}$ to client B.

Client B.

(Optional for key confirmation) Compute $S_2 = {\rm Hash}(0x03 \parallel y_V \parallel {\rm Hash}(x_V \parallel Z_{\rm A} \parallel Z_{\rm B} \parallel x_1 \parallel y_1 \parallel x_2 \parallel y_2))$, and verify if $S_2 = S_{\rm A}$ holds. Terminate if it is not true.

Finally, the session key is established as $K = K_{\rm A} = K_{\rm B}$.

2.2 Lattices

Let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^m$ are n linearly independent vectors. We define an m-dimensional lattice $\mathcal{L}$ as the set of vectors:

$$\mathcal{L} = \left\{ \sum_{i=1}^{n} x_i \boldsymbol{b}_i : x_i \in \mathbb{Z} \right\}.$$

We denote $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ as a basis of $\mathcal{L}$. Without loss of generality, only full-rank (which means the rank n equals the dimension m) lattices are considered in the following of this paper because of the reduction from other cases to full-rank.

In lattice-based cryptography, there are two crucial computational complexity problems known as closest vector problem (CVP) and shortest vector problem (SVP). Although both of them are proved to be NP-hard, their approximate variants will get easier as the increment of the approximate factor, and this is of great importance to practical lattice-based cryptography and cryptanalysis. Next, the definitions of some computational complexity problems and relevant results which are helpful to our discussion are listed as follows.

Definition 1 ([26])**.** Recall the definitions of some lattice-based computational hard problems.

(1) Closest vector problem (CVP): Given a basis of a lattice $\mathcal{L}$ and a target vector $\boldsymbol{t} \in \mathbb{R}^n$, find a lattice vector $\boldsymbol{v}$ which is closest to the target $\boldsymbol{t}$, i.e., ${\rm dist}(\boldsymbol{v}, \boldsymbol{t}) \leqslant {\rm dist}(\boldsymbol{u}, \boldsymbol{t})$ for any vector $\boldsymbol{u} \in \mathcal{L}$, where dist denotes the Euclid norm of two points.

(2) γ-approximate closest vector problem (CVP$_\gamma$): Given a basis of a lattice $\mathcal{L}$ and a target vector $\boldsymbol{t} \in \mathbb{R}^n$, find a lattice vector $\boldsymbol{v}$ such that ${\rm dist}(\boldsymbol{v}, \boldsymbol{t}) \leqslant \gamma {\rm dist}(\boldsymbol{u}, \boldsymbol{t})$ for any lattice vector $\boldsymbol{u} \in \mathcal{L}$.

(3) Shortest vector problem (SVP): Given a basis of a lattice $\mathcal{L}$, find a lattice vector $\boldsymbol{v} \neq \boldsymbol{0}$ such that $\|\boldsymbol{v}\| \leqslant \|\boldsymbol{u}\|$ for any nonzero vector $\boldsymbol{u} \in \mathcal{L}$.

(4) γ-approximate shortest vector problem (SVP$_\gamma$): Given a basis of a lattice $\mathcal{L}$, find a lattice vector $\boldsymbol{v} \neq \boldsymbol{0}$ such that $\|\boldsymbol{v}\| \leqslant \gamma \|\boldsymbol{u}\|$, for any nonzero vector $\boldsymbol{u} \in \mathcal{L}$.

For an n-dimensional lattice, the LLL algorithm [20] can return an approximate shortest lattice vector with approximate factor $(4/3)^{n/2}$. Inherited from the technique of blockwise reduction proposed by Schnorr [27], the BKZ-algorithms [21, 22, 28] have been the most practical algorithm to lattice basis reduction. For the closest vector problem, Babai [29] provided a polynomial-time algorithm to approximate CVP using basis reduction with factor $2^{n/2}$. This result was further improved to $2^{cn \log\log n / \log n}$ on account of the blockwise algorithm and the reduction technique from approximating CVP to approximating SVP [30], which is specified in the following lemma.

Lemma 1 ([8])**.** For any constant $c > 0$, there exists a randomized polynomial-time algorithm which given an n-dimensional lattice L and a vector $\boldsymbol{r} \in \mathbb{R}^n$, finds a vector $\boldsymbol{v} \in L$ satisfying with probability exponentially close to 1 the inequality:

$$\|\boldsymbol{v} - \boldsymbol{r}\| \leqslant 2^{cn \log\log n / \log n} \min\{\|\boldsymbol{z} - \boldsymbol{r}\|,\ \boldsymbol{z} \in L\}.$$

2.3 The hidden number problem (HNP) and extended hidden number problem (EHNP)

The HNP introduced in 1996 by Boneh and Venkatesan [9] is applied to recover the secret key of a DSA-like signature [7,8,11] and their certain implementations such like OpenSSL [19], given some leaked bits of nonces. For any real z and prime n, define the symbol $|\cdot|_n$ as $|z|_n = \min_{b\in\mathbb{Z}}|z-bn|$. For any rational l and m, let $\mathrm{APP}_{l,n}(m)$ denote any rational r such that $|m-r|_n \leqslant \frac{n}{2^{l+1}}$. The HNP can be stated as follows: given many approximation $\mathrm{APP}_{l,n}(\alpha t_i)$ of αt_i for $1 \leqslant i \leqslant d$ where t_i is known and chosen uniformly and randomly from $[1, n-1]$, recover the secret $\alpha \in \mathbb{Z}_n$.

To broaden the usability of HNP to scenarios where the leaked bits distribute discretely, an EHNP was proposed in [18], together with an efficient polynomial time algorithm to solve its instances. The definition of EHNP is as follows.

Definition 2 ([18]). Let N be a prime, and let $x \in \mathbb{Z}_N$ be a particular unknown integer such that

$$x = \bar{x} + \sum_{j=1}^{m} 2^{\pi_j} x_j,$$

where the integers $\bar{x}$ and π_j $(1 \leqslant j \leqslant m)$ are known. The unknown integers x_j satisfy $0 < x_j < 2^{\nu_j}$, where ν_j $(1 \leqslant j \leqslant m)$ are known rational constants. Given d congruences,

$$\alpha_i \sum_{j=1}^{m} 2^{\pi_j} x_j + \sum_{j=1}^{l_i} \rho_{i,j} k_{i,j} \equiv \beta_i - \alpha_i \bar{x} \pmod{N}, \quad 1 \leqslant i \leqslant d,$$

where $\alpha_i \neq 0 \pmod{N}$, π_j, $\rho_{i,j}$ and β_i are known values. The unknown integers $k_{i,j}$ satisfy $0 \leqslant k_{i,j} \leqslant 2^{\mu_{i,j}}$, where $\mu_{i,j}$ are known. The EHNP is to find x (the hidden number) and its instance is represented by

$$(\bar{x}, N, \{\pi_j, \nu_j\}_{j=1}^{m}, \{\alpha_i, \{\rho_{i,j}, \mu_{i,j}\}_{j=1}^{l_i}, \beta_i\}_{i=1}^{d}).$$

In the following of this paper, for $x \in \mathbb{Z}$ and $k > 0$, we denote $\mathrm{LSB}_k(x)$ as the integer $h \in [0, 2^k)$ such that $h = x \bmod 2^k$, and $\mathrm{MSB}_k(x)$ as the integer $s > 0$ such that $s = (x - x \bmod 2^{\mathrm{length}(x)-k})/2^{\mathrm{length}(x)-k}$, where $\mathrm{length}(x)$ denotes the binary length of x.

3 Partially known information attack on SM2 key exchange protocol

In the SM2 key exchange protocol, the idea to recover the private key d_A of user A is to use the congruence,

$$t_\mathrm{A} = (d_\mathrm{A} + \overline{x}_1 \cdot r_\mathrm{A}) \bmod n, \tag{1}$$

where r_A is randomly chosen from $[1, n-1]$ and $\overline{x}_1$ is a calculable value related to r_A.

Obviously, if r_A is leaked, then the ability to recover d_A will be bounded by the extend of t_A's disclosure due to the deterministic linear translation relationship between d_A and t_A. Nevertheless, it is an HNP-like issue if t_A is totally known as well as some least significant bits of the nonce r_A. Suppose that $r_\mathrm{A} = 2^l \cdot b + a$ with a leaked, rewrite the above congruence as

$$-(2^l \overline{x}_1)^{-1} d_\mathrm{A} - 2^{-l} a + (2^l \overline{x}_1)^{-1} t_\mathrm{A} = b \bmod n,$$

where $b \in [0, n/2^l)$. Then recovering the secret key d_A is therefore a generalized HNP provided that the coefficient $(2^l \overline{x}_1)^{-1}$ is sufficiently uniform to make the corresponding HNP provably tractable.

However, there are two obstacles in approaching HNP in the case of SM2 key exchange protocol. The first one is that t_A is private which is transformed from the private key by a perturbation of some random nonce r_A. The second one is the randomness of $t_\mathrm{A} - d_\mathrm{A}$, and this makes it impossible to take $t_\mathrm{A} - d_\mathrm{A}$ as unitary. As a result, it is reasonable to consider the cases in which partial information of r_A and t_A are leaked, and this approach is much more preferable in practice from the perspective of side channel attack. In this section, utilizing the similar technique of EHNP which is based on the lattice basis reduction, we give a partially known information attack on SM2 key exchange protocol, by considering the LSBs and MSBs leakage separately.

3.1 Recover LSBs of private key with LSBs leakage of nonces

In this subsection, we consider the following scenario. Given a number of key exchange instances with known $\overline{x}_1$, say m of them, we shall show that partially known LSBs of r_{A} and t_{A} will disclose some LSBs of the private key d_{A}. For simplicity, we rewrite congruences (1) as

$$t_i = (d_{\mathrm{A}} + \overline{x}_i \cdot r_i) \bmod n,$$

for $1 \leqslant i \leqslant m$. If the l least significant bits of r_i and the l' least significant bits of t_i are leaked (where $1 \leqslant l \leqslant \mathrm{length}(r_i)$ and $1 \leqslant l' \leqslant \mathrm{length}(t_i)$), that is

$$r_i = 2^l \cdot b_i + a_i, \quad t_i = 2^{l'} \cdot e_i + f_i, \quad 1 \leqslant i \leqslant m.$$

Here $a_i \in [0, 2^l - 1]$ and $f_i \in [0, 2^{l'} - 1]$ are known LSBs of r_i and t_i respectively. $b_i \in [0, n/2^l)$ and $e_i \in [0, n/2^{l'})$ are unknown. Rearranging the congruences, we have

$$d_{\mathrm{A}} + 2^l \overline{x}_i b_i - 2^{l'} e_i = f_i - \overline{x}_i a_i \bmod n, \quad 1 \leqslant i \leqslant m.$$

The solutions to the system of the above equations could be represented by points from the lattice as follows. Define a $(3m+1)$-dimensional lattice L spanned by the rows of the following matrix:

$$\boldsymbol{B} = \begin{pmatrix} n\mathrm{I}_m & & & & & & & & \\ 1 & \cdots & & 1 & \frac{\delta}{n} & & & & \\ 2^l\overline{x}_1 & & & & \frac{2^l\delta}{n} & & & & \\ -2^{l'} & & & & & \frac{2^{l'}\delta}{n} & & & \\ & 2^l\overline{x}_2 & & & & & & & \\ & -2^{l'} & & & & & & & \\ & & \ddots & & & & \ddots & & \\ & & & 2^l\overline{x}_m & & & & \frac{2^l\delta}{n} & \\ & & & -2^{l'} & & & & & \frac{2^{l'}\delta}{n} \end{pmatrix},$$

where δ is a parameter to be determined later. We aim to transform the problem to CVP_γ on the lattice L.

Lemma 2. In SM2 key exchange protocol, $\{\overline{x}_i\}_{i=1}^m$ are independently and uniformly distributed from $[\sqrt{n}/2, \sqrt{n}]$.

Proof. First, we review the choices of $\{\overline{x}_i\}_{i=1}^m$. They are generated during independently invoking SM2 key exchange protocol repeatedly for m times. In the i-th key exchange procedure, client A randomly chooses an integer $r_i \in [1, n-1]$ and calculates $(x_i, y_i) = r_i G$, where G is the fixed point of order n, then sends (x_i, y_i) to client B. $\overline{x}_i$ is computed as $2^w + (x_i \& (2^w - 1))$ with $w = \lceil \lceil \log n \rceil / 2 \rceil - 1$. Clearly, x_i is independently and uniformly distributed due to the randomness of r_i, and this is true for $\overline{x}_i$. According to the definition of $\overline{x}_i$, we have

$$2^w < \overline{x}_i < 2^w + (2^w - 1) = 2^{w+1} - 1.$$

It is clear that $2^{w+1} \simeq \sqrt{n}$. Hence, $\overline{x}_i$ is distributed from $[\sqrt{n}/2, \sqrt{n}]$.

The following lemma formalizes the characteristic of any lattice point when it is short enough in l_∞ norm. The relationship between the short lattice vector and the private key stated in this lemma is crucial to the success probability of our algorithm.

Lemma 3. For any lattice vector $\boldsymbol{r} = \boldsymbol{z}\boldsymbol{B} = (r_1, \ldots, r_{3m+1}) \in L$ with the coordinates vector $\boldsymbol{z} = (e_1, \ldots, e_m, y, t_{1,1}, t_{1,2}, \ldots, t_{i,1}, t_{i,2}, \ldots, t_{m,1}, t_{m,2})$ satisfying $\|\boldsymbol{r}\|_\infty < \kappa\delta$,

$$y = 0 \bmod n \bmod 2^{l'}$$

holds with probability higher than $1 - \frac{\kappa^{2m+1} n^{m+1}}{2^{(l+l'-2)m}}$.

Proof. Since $\|\boldsymbol{r}\|_\infty < \kappa\delta$, we have

$$\begin{aligned}
&|r_i| = |ne_i + y + 2^l\overline{x}_i t_{i,1} - 2^{l'} t_{i,2}| < \kappa\delta < 1, \quad \text{for } 1 \leqslant i \leqslant m,\\
&|r_{m+1}| = \left|\frac{\delta}{n} y\right| < \kappa\delta,\\
&|r_{m+2i}| = \left|\frac{2^l\delta}{n} t_{i,1}\right| < \kappa\delta < 1, \quad \text{for } 1 \leqslant i \leqslant m,\\
&|r_{m+2i+1}| = \left|\frac{2^{l'}\delta}{n} t_{i,2}\right| < \kappa\delta < 1, \quad \text{for } 1 \leqslant i \leqslant m.
\end{aligned}$$

Notice that $\boldsymbol{r}$ is an integer vector, then

$$2^l\overline{x}_i t_{i,1} - 2^{l'} t_{i,2} = -y \bmod n, \quad \text{for } 1 \leqslant i \leqslant m, \tag{2}$$

$$|y| < \kappa n, \tag{3}$$

$$|t_{i,1}| < \frac{\kappa n}{2^l}, \quad |t_{i,2}| < \frac{\kappa n}{2^{l'}}, \quad \text{for } 1 \leqslant i \leqslant m. \tag{4}$$

Assume the following relations for $1 \leqslant i \leqslant m$,

$$\begin{aligned}
&y = \kappa_y n + y_0 \text{ with } |\kappa_y| < \kappa \text{ and } 0 \leqslant y_0 < n,\\
&2^{l'} t_{i,2} = \kappa_i n + 2^{l'} t'_{i,2} + \epsilon \text{ with } 0 < t'_{i,2} < n/2^{l'}, \quad |\kappa_i| < \kappa \text{ and } 0 < \epsilon < 2^{l'},\\
&t'_{i,1} = t_{i,1} - (2^l\overline{x}_i)^{-1}\ \epsilon \bmod n,\\
&(2^l\overline{x}_i)\cdot(2^l\overline{x}_i)^{-1} = 1 + \kappa'_i n,\\
&e'_i = e_i + \kappa_y - \kappa_i + \kappa'_i\epsilon.
\end{aligned}$$

Then the vector $\boldsymbol{r}' = \boldsymbol{z}'\boldsymbol{B}$ is also a lattice point with coordinates vector $\boldsymbol{z} = (e'_1, \ldots, e'_m, y_0, t'_{1,1}, t'_{1,2}, \ldots, t'_{i,1}, t'_{i,2}, \ldots, t'_{m,1}, t'_{m,2})$, which satisfies

$$\begin{aligned}
|r'_i| &= |ne'_i + y_0 + 2^l\overline{x}_i t'_{i,1} - 2^{l'} t'_{i,2}|\\
&= |ne'_i + y - \kappa_y n + 2^l\overline{x}_i(t_{i,1} - (2^l\overline{x}_i)^{-1}\ \epsilon) + \kappa_i n - 2^{l'} t_{i,2} + \epsilon|\\
&= |n(e'_i - \kappa_y - \kappa'_i\epsilon + \kappa_i) + y + 2^l\overline{x}_i t_{i,1} - 2^{l'} t_{i,2}|\\
&= |ne_i + y + 2^l\overline{x}_i t_{i,1} - 2^{l'} t_{i,2}|\\
&= |r_i| < \kappa\delta < 1, \quad \text{for } 1 \leqslant i \leqslant m.
\end{aligned}$$

This implies that $ne'_i + y_0 + 2^l\overline{x}_i t'_{i,1} - 2^{l'} t'_{i,2} = 0$. Therefore, we obtain

$$2^l\overline{x}_i t'_{i,1} - 2^{l'} t'_{i,2} = -y_0 \bmod n, \quad \text{for } 1 \leqslant i \leqslant m. \tag{5}$$

For some fixed y, consider the event E defined as $2^{l'} t'_{i,2} - y_0 \neq 0 \bmod n$ for all $1 \leqslant i \leqslant m$. Notice that, E implies that Eq. (2) has a non-trivial short solution on $(t_{i,1}, t_{i,2})$. To prove the upper bound for the probability that E happens, we first evaluate the probability that Eq. (5) has a non-trivial solution on $\{(t_{i,1}, t_{i,2}) \mid |t_{i,1}| < \frac{\kappa n}{2^l}$ and $|t_{i,2}| < \frac{\kappa n}{2^{l'}}$ for $1 \leqslant i \leqslant m\}$, given that $\overline{x}_i$ is uniformly and independently distributed on $[\sqrt{n}/2, \sqrt{n}]$ according to Lemma 2.

Clearly, there exist at most $(2\kappa n/2^l - 1)\cdot(2\kappa n/2^{l'} - 1)$ number of possible tuples $(t_{i,1}, t_{i,2})$ such that $\overline{x}_i = (2^l t'_{i,1})^{-1}(2^{l'} t'_{i,2} - y_0) \bmod n$ is non-zero. Considering the uniform distribution of $\overline{x}_i$ on $[\sqrt{n}/2, \sqrt{n}]$, the probability that Eq. (5) has a non-zero solution on $(t_{i,1}, t_{i,2})$ is less than

$$\begin{aligned}
P_i(y) &= \frac{(2\kappa n/2^l - 1)\cdot(2\kappa n/2^{l'} - 1)\cdot\sqrt{n}/2n}{\sqrt{n}/2}\\
&\leqslant \frac{\kappa^2 n}{2^{l+l'-2}}.
\end{aligned}$$

Because all $\{\overline{x}_i\}_{i=1}^m$ are independent, the probability that E happens under the condition of fixed y is less than

$$\prod_{i=1}^{m} P_i(y) \leqslant \frac{(\kappa^2 n)^m}{2^{(l+l'-2)m}}.$$

Denote P_E as the probability that E happens. With the bound for y from (3), we obtain

$$P_E \leqslant \kappa n \cdot \frac{(\kappa^2 n)^m}{2^{(l+l'-2)m}} = \frac{\kappa^{2m+1} n^{m+1}}{2^{(l+l'-2)m}}.$$

Consequently, there exists some $w \in [1, m]$ such that

$$2^{l'} t'_{w,2} - y_0 = 0 \bmod n$$

holds with probability $1 - P_E$. Correspondingly, we obtain

$$y = 0 \bmod n \bmod 2^{l'}$$

with probability $1 - P_E$, and this concludes the proof.

Theorem 1. For SM2 key exchange protocol with the private key of Client A denoted as d_{A}, there exists a polynomial time algorithm which returns $\mathrm{LSB}_{l'}(d_{\mathrm{A}})$ with probability at least $1 - \frac{\kappa^{2m+1} n^{m+1}}{2^{(l+l'-2)m}}$, with knowledge of $\mathrm{LSB}_l(r_i)$ and $\mathrm{LSB}_{l'}(t_i)$ $(1 \leqslant i \leqslant m)$ for m instances. Here, l and l' are integers smaller than the length of recommended SM2 module parameter and $\kappa = (1 + 2^{(3m+1)\log\log(3m+1)/\log(3m+1)}\sqrt{2m+1})/2$.

Proof. For the sake of completeness, we briefly review the basics of exploiting the leaked information during the protocol process. Given a sequence of random $\{\overline{x}_i\}_{i=1}^m$, $\mathrm{LSB}_l(r_i)$ and $\mathrm{LSB}_{l'}(t_i)$, denoted by a_i and f_i respectively, we have

$$d_{\mathrm{A}} + 2^l \overline{x}_i b_i - 2^{l'} e_i = f_i - \overline{x}_i a_i \bmod n, \quad 1 \leqslant i \leqslant m,$$

where d_{A} is the private key of client A, $0 \leqslant b_i < n/2^l$, and $0 \leqslant\ e_i < n/2^{l'}$.

Denote $\beta_i = f_i - \overline{x}_i a_i$ for $1 \leqslant i \leqslant m$. Define the target vector

$$\boldsymbol{v} = \left(\beta_1, \beta_2, \ldots, \beta_m, \frac{\delta}{2}, \frac{\delta}{2}, \ldots, \frac{\delta}{2}\right) \in \mathbb{R}^{3m+1}.$$

There exists a lattice vector

$$\boldsymbol{u} = \boldsymbol{h}\boldsymbol{B} = \left(\beta_1, \beta_2, \ldots, \beta_m, d_{\mathrm{A}}\frac{\delta}{n}, b_1\frac{2^l\delta}{n}, e_1\frac{2^{l'}\delta}{n}, \ldots, \ldots, b_m\frac{2^l\delta}{n}, e_m\frac{2^{l'}\delta}{n}\right) \in\ L,$$

with coordinates vector $\boldsymbol{h} = (c_1, \ldots, c_m, d_{\mathrm{A}}, b_1, e_1, b_2, e_2, \ldots, \ldots, b_m, e_m) \in \mathbb{Z}^{3m+1}$.
Then

$$\boldsymbol{u} - \boldsymbol{v} = \left(0, \ldots, 0, \frac{\delta d_{\mathrm{A}}}{n} - \frac{\delta}{2}, b_1\frac{2^l\delta}{n} - \frac{\delta}{2}, e_1\frac{2^{l'}\delta}{n} - \frac{\delta}{2}, \ldots, b_m\frac{2^l\delta}{n} - \frac{\delta}{2}, e_m\frac{2^{l'}\delta}{n} - \frac{\delta}{2}\right).$$

It is clear that

$$-\frac{\delta}{2} < \frac{\delta}{n} - \frac{\delta}{2} \leqslant \frac{\delta d_{\mathrm{A}}}{n} - \frac{\delta}{2} \leqslant \delta - \frac{\delta}{2} = \frac{\delta}{2},$$

which leads to

$$\left|\frac{\delta d_{\mathrm{A}}}{n} - \frac{\delta}{2}\right| < \frac{\delta}{2}.$$

We also have

$$-\frac{\delta}{2} < b_i\frac{2^l\delta}{n} - \frac{\delta}{2} < \frac{n}{2^l}\frac{2^l\delta}{n} - \frac{\delta}{2} = \frac{\delta}{2},$$

and

$$-\frac{\delta}{2} < e_i\frac{2^{l'}\delta}{n} - \frac{\delta}{2} < \frac{n}{2^{l'}}\frac{2^{l'}\delta}{n} - \frac{\delta}{2} = \frac{\delta}{2},$$

for all $1 \leqslant i \leqslant m$. Hence, we get

$$\|\boldsymbol{u} - \boldsymbol{v}\|_\infty < \frac{\delta}{2}.$$

Furthermore,

$$\|\boldsymbol{u} - \boldsymbol{v}\| < \frac{\delta}{2}\sqrt{2m+1}.$$

For the target vector $\boldsymbol{v}$, there exist a polynomial-time algorithm according to Lemma 1 which can find a lattice vector $\boldsymbol{w} \in L$ satisfying

$$\begin{aligned}\|\boldsymbol{v} - \boldsymbol{w}\| &\leqslant 2^{D \log\log D/\log D}\mathrm{dist}(\boldsymbol{v}, L)\\ &\leqslant 2^{D \log\log D/\log D}\|\boldsymbol{v} - \boldsymbol{u}\|\\ &\leqslant \frac{\delta}{2}2^{D \log\log D/\log D}\sqrt{2m+1},\end{aligned}$$

where $D = 3m + 1$ is the dimension of L.

Let $\boldsymbol{\Delta} = \boldsymbol{u} - \boldsymbol{w}$, then

$$\begin{aligned}\|\boldsymbol{\Delta}\|_\infty &\leqslant \|\boldsymbol{u} - \boldsymbol{v}\|_\infty + \|\boldsymbol{v} - \boldsymbol{w}\|_\infty\\ &\leqslant \|\boldsymbol{u} - \boldsymbol{v}\|_\infty + \|\boldsymbol{v} - \boldsymbol{w}\|\\ &\leqslant \frac{\delta}{2}(1 + 2^{D \log\log D/\log D}\sqrt{2m+1}).\end{aligned}$$

Let $\kappa = (1 + 2^{D \log\log D/\log D}\sqrt{2m+1})/2$, and choose $\delta > 0$ such that $\kappa\delta < 1$. Denote $\boldsymbol{w} = (c'_1, \ldots, c'_m, d', k_{1,1}, k_{1,2}, \ldots, k_{m,1}, k_{m,2})\boldsymbol{B}$. Then according to Lemma 3, with probability larger than $1 - \frac{\kappa^{2m+1}n^{m+1}}{2^{(l+l'-2)m}}$, the $(m+1)$-th coordinate of the representation of $\boldsymbol{\Delta}$ under basis $\boldsymbol{B}$ is 0 module $2^{l'}$. Equivalently, we get

$$d' \bmod n \bmod 2^{l'} = d_{\mathrm{A}} \bmod 2^{l'}.$$

It is noticed that $d' < n$ generally holds since the lattice vector $\boldsymbol{w}$ close to target vector $\boldsymbol{v}$ has been reduced iteratively. So we have $\mathrm{LSB}_{l'}(d') = \mathrm{LSB}_{l'}(d_{\mathrm{A}})$. Hence, finding $\boldsymbol{w}$ discloses $\mathrm{LSB}_{l'}(d_{\mathrm{A}})$. Moreover, the solution elaborated above could be referred to Algorithm 1.

Algorithm 1 Partially known information attack on SM2 key exchange protocol

Input: An integer m, $\{\mathrm{LSB}_l(r_i)\}_{i=1}^m$, $\{\mathrm{LSB}_{l'}(t_i)\}_{i=1}^m$, $\{\overline{x}_i\}_{i=1}^m$;
Output: $\mathrm{LSB}_{l'}(d_{\mathrm{A}})$, where d_{A} is the private key of client A;
1: $\kappa \leftarrow (1 + 2^{(3m+1)\log\log(3m+1)/\log(3m+1)}\sqrt{2m+1})/2$;
2: Select $\delta > 0$ such that $\kappa\delta < 1$;
3: Compute $\beta_i = \mathrm{LSB}_{l'}(t_i) - \overline{x}_i\mathrm{LSB}_l(r_i)$;
4: $\boldsymbol{v} \leftarrow (\beta_1, \beta_2, \ldots, \beta_m, \frac{\delta}{2}, \frac{\delta}{2}, \ldots, \frac{\delta}{2}) \in \mathbb{R}^{3m+1}$;
5: Call algorithm from Lemma 1 to obtain a lattice vector $\boldsymbol{w} \in L$ with $\boldsymbol{w} = (c'_1, \ldots, c'_m, d', k_{1,1}, k_{1,2}, \ldots, k_{m,1}, k_{m,2})\boldsymbol{B}$, which is close to $\boldsymbol{v}$;
6: Return $d' \bmod n \bmod 2^{l'}$.

Additionally, we indicate that one cannot recover the complete private key in this scenario, only with the leakage of $\mathrm{LSB}_l(r_i)$ and $\mathrm{LSB}_{l'}(t_i)$ $(1 \leqslant i \leqslant m)$. Indeed, $\mathrm{LSB}_{l'}(d_{\mathrm{A}})$ is the optimal result becasue there exist plenty of solutions on the tuple (d_{A}, t, r) which coincide with the leakage, and there is no sufficient information to distinguish the right one. The maximum size of the least significant bits of d_{A} which can be revealed is bounded by t'_is leakage.

3.2 Recover MSBs of private key with MSBs leakage of nonces

Based on the analysis of Subsection 3.1, a similar argument works for the case of MSBs leaks. Suppose that the $\mathrm{LSB}_{l'}(d_{\mathrm{A}})$ has been recovered, we shall show how to recover the remainder part of the private key with MSBs leakage of nonces.

Let $d_{\mathrm{A}} = 2^{l'}d_1 + d_0$, $r_i = 2^l \cdot b_i + a_i$, and $t_i = 2^{l'} \cdot e_i + f_i$ $(1 \leqslant i \leqslant m)$ with b_i, e_i known and d_0 recovered. The relationship $t_i = (d_{\mathrm{A}} + \overline{x}_i \cdot r_i) \bmod n$ holds among these variables, which could be rewritten as

$$2^{l'}d_1 + \overline{x}_i a_i - f_i = 2^{l'}e_i - 2^l \overline{x}_i b_i - d_0 \bmod n, \quad 1 \leqslant i \leqslant m.$$

Similarly, we define a $(3m+1)$-dimensional lattice Λ spanned by the rows of the following matrix:

$$\boldsymbol{D} = \begin{pmatrix} n\mathrm{I}_m & & & & & & & \\ 2^{l'} & \cdots & 2^{l'} & \frac{2^{l'}\eta}{n} & & & & \\ \overline{x}_1 & & & & \frac{\eta}{2^l} & & & \\ -1 & & & & & \frac{\eta}{2^{l'}} & & \\ & \overline{x}_2 & & & & & & \\ & -1 & & & & & & \\ & & \ddots & & & & \ddots & \\ & & & \overline{x}_m & & & & \frac{\eta}{2^l} \\ & & & -1 & & & & \frac{\eta}{2^{l'}} \end{pmatrix},$$

where η is a parameter to be determined later.

Lemma 4. For any lattice vector $\boldsymbol{r} = \boldsymbol{z}\boldsymbol{D} = (r_1, \ldots, r_{3m+1}) \in \Lambda$ with coordinates vector $\boldsymbol{z} = (e_1, \ldots, e_m, y, t_{1,1}, t_{1,2} \ldots, t_{i,1}, t_{i,2}, \ldots, t_{m,1}, t_{m,2})$ satisfying $\|\boldsymbol{r}\|_\infty < \kappa\eta$,

$$2^{l'}y \bmod n = (2^{l'}y \bmod n) \bmod 2^{l'}$$

holds with probability at least $1 - \frac{\kappa^{2m+1}2^{(l+l'+2)m-l'}}{n^{m-1}}$.

One can refer to Appendix A for the proof of Lemma 4 which is similar to that of Lemma 3.

Theorem 2. There exists a polynomial time algorithm which returns $\mathrm{MSB}_{l'}(d_{\mathrm{A}})$ with probability larger than $1 - \frac{\kappa^{2m+1}2^{(l+l'+2)m-l'}}{n^{m-1}}$ where d_{A} is the private key in SM2 key exchange protocol, given $\mathrm{MSB}_l(r_i)$ and $\mathrm{MSB}_{l'}(t_i)$ $(1 \leqslant i \leqslant m)$ leakage of m instances.

Proof. For a sequence of m instances of SM2 key exchange protocol with the same private key $t_i = (d_{\mathrm{A}} + \overline{x}_i \cdot r_i) \bmod n$ $(1 \leqslant i \leqslant m)$, suppose that the private key $d_{\mathrm{A}} = 2^{l'}d_1 + d_0$, $r_i = 2^l \cdot b_i + a_i$, and $t_i = 2^{l'} \cdot e_i + f_i$ $(1 \leqslant i \leqslant m)$ with b_i, e_i known and d_0 recovered. Then we have

$$2^{l'}d_1 + \overline{x}_i a_i - f_i = 2^{l'}e_i - 2^l \overline{x}_i b_i - d_0 \bmod n, \quad 1 \leqslant i \leqslant m.$$

Now we aim to recover d_1 based on the similar idea of Theorem 1 and the definitions are consistent essentially.

Denote $\beta_i = 2^{l'}e_i - 2^l\overline{x}_i b_i - d_0 \bmod n$ for $1 \leqslant i \leqslant m$. Define the target vector $\boldsymbol{v} = (\beta_1, \beta_2, \ldots, \beta_m, \frac{\eta}{2}, \frac{\eta}{2}, \ldots, \frac{\eta}{2}) \in \mathbb{R}^{3m+1}$. Then, there exists a lattice vector $\boldsymbol{u} = \boldsymbol{h}\boldsymbol{D} = (\beta_1, \beta_2, \ldots, \beta_m, d_1\frac{2^{l'}\eta}{n}, a_1\frac{\eta}{2^l}, f_1\frac{\eta}{2^{l'}}, \ldots, \ldots, a_m\frac{\eta}{2^l}, f_m\frac{\eta}{2^{l'}}) \in \Lambda$, with coordinates vector $\boldsymbol{h} = (c_1, \ldots, c_m, d_1, a_1, f_1, a_2, f_2, \ldots, \ldots, a_m, f_m) \in \mathbb{Z}^{3m+1}$. Their distance in l_∞ norm is bounded by

$$\begin{aligned} \|\boldsymbol{u} - \boldsymbol{v}\|_\infty &= \left\| \left(0, \ldots, 0, d_1\frac{2^{l'}\eta}{n} - \frac{\eta}{2}, a_1\frac{\eta}{2^l} - \frac{\eta}{2}, f_1\frac{\eta}{2^{l'}} - \frac{\eta}{2}, \ldots, a_m\frac{\eta}{2^l} - \frac{\eta}{2}, f_m\frac{\eta}{2^{l'}} - \frac{\eta}{2}\right) \right\|_\infty \\ &\leqslant \frac{\eta}{2}, \end{aligned}$$

A lattice vector $\boldsymbol{w} \in \Lambda$ could be found in polynomial time by the algorithm provided in Lemma 1 which is close to the target vector $\boldsymbol{v}$ by

$$\begin{aligned} \|\boldsymbol{v} - \boldsymbol{w}\| &\leqslant 2^{D \log\log D / \log D}\mathrm{dist}(\boldsymbol{v}, \Lambda) \\ &\leqslant \frac{\eta}{2}2^{D \log\log D / \log D}\sqrt{2m+1}, \end{aligned}$$

where $D = 3m + 1$ is the dimension of Λ.

Denote $\boldsymbol{\Delta} = \boldsymbol{u} - \boldsymbol{w}$, then

$$\begin{aligned}\|\boldsymbol{\Delta}\|_\infty &\leqslant \|\boldsymbol{u} - \boldsymbol{v}\|_\infty + \|\boldsymbol{v} - \boldsymbol{w}\|_\infty \\ &\leqslant \|\boldsymbol{u} - \boldsymbol{v}\|_\infty + \|\boldsymbol{v} - \boldsymbol{w}\| \\ &\leqslant \frac{\eta}{2}(1 + 2^{D \log\log D/\log D}\sqrt{2m+1}).\end{aligned}$$

Let $\kappa = (1 + 2^{D \log\log D/\log D}\sqrt{2m+1})/2$, and choose $\eta > 0$ such that $\kappa\eta < 1$. Denote $\boldsymbol{w} = (c'_1, \ldots, c'_m, d', k_{1,1}, k_{1,2}, \ldots, k_{m,1}, k_{m,2})\boldsymbol{\Lambda}$. Then according to Lemma 4, with probability larger than $1 - \frac{\kappa^{2m+1} 2^{(l+l'+2)m-l'}}{n^{m-1}}$, the $(m+1)$-th coordinate of the representation of $\boldsymbol{\Delta}$ under basis $\boldsymbol{\Lambda}$ satisfies

$$2^{l'}(d' - d_1) \bmod n = (2^{l'}(d' - d_1) \bmod n) \bmod 2^{l'},$$

which results in

$$d_1 = (2^{l'} d' \bmod n - (2^{l'} d' \bmod n) \bmod 2^{l'})/2^{l'}.$$

Indeed, $2^{l'} d'$ is usually smaller than n due to the properties of lattice basis reduction during the way to find $\boldsymbol{w}$, and this leads to

$$d_1 = d'.$$

4 Experiments

In this section, we report our experimental results on partially known information attack to SM2 key exchange protocol. Since the treatment of MSBs is similar to that of LSBs, we just list the results to recover the private key with LSBs leakage of nonces.

We implement our attack provided in Section 3 repeatedly. All executions performed on an Intel Core i7-6700 CPU running at 3.40 GHz and all the codes are written in C++. We invoke the BKZ algorithm in NTL library [31] and use the embedding strategy [32] instead of Babai's nearest plane algorithm to disclose the hidden private key, because the embedding technique usually performs much better to solve CVP in practice. Indeed, it is a method to reduce the bounded distance CVP to SVP. Given a lattice with basis $B = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n]$ and a target vector $\boldsymbol{v} \in \mathrm{span}(B)$, the embedding technique constructs a new $(n+1)$-dimensional lattice spanned by the row vectors $(\boldsymbol{b}_i, 0)$ $(1 \leqslant i \leqslant n)$ and $(\boldsymbol{v}, \beta)$, where β is a parameter to be determined. In our experiments, we take β as $\delta/2$ (or $\eta/2$ respectively) to balance the coefficients of the lattice. The success of the strategy is guaranteed by the possibility that the shortest vector from the reduced basis of the embedding lattice is of the form $(\boldsymbol{v} - \boldsymbol{a}, \beta)$ where $\boldsymbol{a}$ is a lattice point sufficiently close to the target $\boldsymbol{v}$. This form of short lattice vector usually happens on the second or third vector of the reduced basis in our experiments.

To verify the correctness of Theorem 1 and explore the asymptotic lower bound for the size of leakage, we first simulate the cases of 64-bits n and 64-bits p and the results are shown in Figure 1(a). The results on small size reveal that the success probability will be considerably large, if there are enough leaked bits in each trial. Similarly, to determine the success boundary, for the set of parameters in SM2 document [1] where n and p are recommended to be 256 bits, we do experiments on instances with $260 \leqslant l + l' \leqslant 280$, $80 \leqslant l \leqslant 180$ and $80 \leqslant l' \leqslant 180$. Our experiments show that the sum of known bits is crucial to success probability, and this coincides with the result of Theorem 1. Since the existing lattice reduction algorithms usually perform better in practice, the experiment results are much better than theoretical analysis as shown in Theorem 1. Given several bits more than half of all the secrete intermediate values in each instance, the corresponding part of the private key could be extracted over about 30 instances with considerable success probability. The results are displayed in Figure 1(b). Specifically, when the known fraction are more than 256 bits which means $l + l' \geqslant 265$, the success probability can achieve 0.95 under the optimization of l and l'. When l and l' are set to be 179 and 80 separately such that the number of

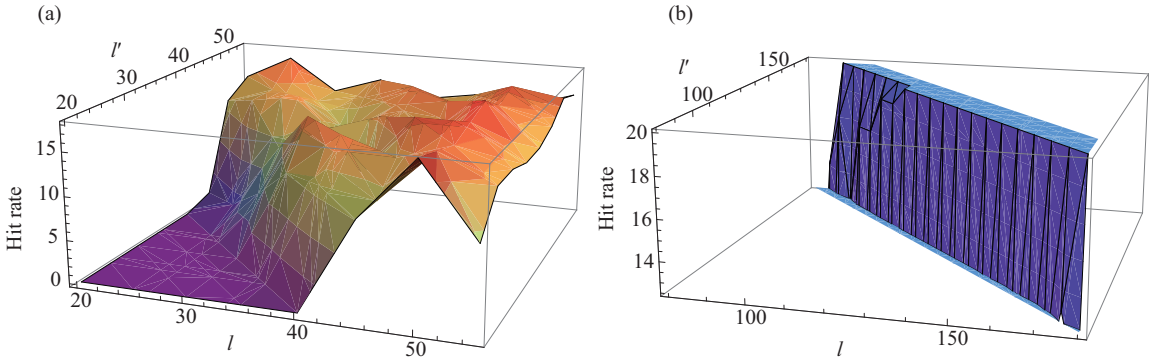

Figure 1 (Color online) The hit rate of attacks on 20 random instances, each derived simulatively from a leakage of the SM2 key exchange protocol with parameter size (a) 64 bits and (b) 256 bits.

known bits is 259, our experiments show that the attack can succeed with overwholeming probability over 40 samples. Additionally, note that our algorithm may perform better by further optimizations invoking new techniques from [23–25].

5 Conclusion

In this paper, we give the partially known information attack on SM2 key exchange protocol. We propose a polynomial-time algorithm which provably recover the user's secret key when some bits of the intermediate variables are leaked. Due to the better practical performance of lattice reduction algorithms, the algorithm succeeds in our experiments possessing larger success probability with the same size of leakage than theoretical results. Specifically, we show that, given about half least significant bits of the two unknown intermediate values in each congruence over about 30 instances, the corresponding secret key could be revealed with overwhelming probability. Inspired by our attack, to resist against side-channel information leaks, countermeasures such like masking technique should be adopted to randomize the intermediate values in the implementation of SM2 key exchange protocol.

Acknowledgements This work was supported by National Key Research and Development Program of China (Grant Nos. 2016YFB0800902, 2016YFF0204004), and National Nature Science Foundation of China (Grant No. 61402536). The authors would like to thank the anonymous referees for their valuable comments.

References

1 Office of State Commercial Cryptography Administration. Public key cryptographic alforithm SM2 based on elliptic curves (in chinese). 2010. http://www.oscca.gov.cn/UpFile/2010122214822692.pdf
2 International Organization for Standardization. Information technology, trusted platform module library, Part 1: Architecture. ISO/IEC 11889-1:2015. https://www.iso.org/standard/66510.html
3 International Organization for Standardization. Information technology, security techniques digital signatures with appendix Part 3: discrete logarithm based mechanisms. ISO/IEC 14888-3:2016. https://www.iso.org/standard/64267.html
4 Diffie W, Hellman M E. New directions in cryptography. IEEE Trans Inform Theor, 1976, 22: 644–654
5 Howgrave-Graham N A, Smart N P. Lattice attacks on digital signature schemes. Dess Codes Cryptography, 2001, 23: 283–290
6 Liu M, Nguyen P Q. Solving BDD by enumeration: an update. In: Topics in Cryptology–CT-RSA 2013. Berlin: Springer, 2013. 7779: 293–309
7 Nguyen P Q. The dark side of the hidden number problem: lattice attacks on DSA. In: Cryptography and Computational Number Theory. Basel: Birkhäuser, 2001. 321–330
8 Nguyen P Q, Shparlinski I E. The insecurity of the digital signature algorithm with partially known nonces. J Cryptology, 2002, 15: 151–176
9 Boneh D, Venkatesan R. Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes. In: Advances in Cryptology–CRYPTO'96. Berlin: Springer, 1996. 1109: 129–142
10 Shani B. On the bit security of elliptic curve Diffie-Hellman. In: Proceedings of the 20th IACR International Conference on Practice and Theory in Public-Key Cryptography. Berlin: Springer, 2017. 10174: 361–387

11 Liu M, Chen J, Li H. Partially known nonces and fault injection attacks on SM2 signature algorithm. In: Proceedings of the 9th International Conference on Information Security and Cryptology. Cham: Springer, 2013. 8567: 343–358
12 Aranha D F, Fouque P A, Gérard B, et al. GLV/GLS decomposition, power analysis, and attacks on ECDSA signatures with single-bit nonce bias. In: Advances in Cryptology–ASIACRYPT 2014. Berlin: Springer, 2014. 8873: 262–281
13 Bleichenbacher D. On the generation of one-time keys in DL signature schemes. In: Proceedings of IEEE P1363 Working Group Meeting, 2000
14 Bleichenbacher D. On the generation of dsa one-time keys. In: Presentation at Cryptography Research Inc., 2007
15 De Mulder E, Hutter M, Marson M E, et al. Using Bleichenbacher's solution to the hidden number problem to attack nonce leaks in 384-bit ECDSA. In: Cryptographic Hardware and Embedded Systems–CHES 2013. Berlin: Springer, 2013. 8086: 435–452
16 Boneh D, Halevi S, Howgrave-Graham N A. The modular inversion hidden number problem. In: Advances in Cryptology–ASIACRYPT 2001. Berlin: Springer, 2001. 2248: 36–51
17 Shparlinski I E. Playing hide-and-seek with numbers: the hidden number problem, lattices and exponential sums. In: Proceedings of Symposia in Applied Mathematics, 2005. 62: 153–177
18 Hlaváč M, Rosa T. Extended hidden number problem and its cryptanalytic applications. In: Selected Areas in Cryptography: 13th International Workshop, SAC 2006. Berlin: Springer, 2006. 4356: 114–133
19 Fan S, Wang W, Cheng Q. Attacking OpenSSL implementations of ECDSA with a few signatures. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2016. 1505–1515
20 Lenstra A K, Lenstra H W, Lovász L. Factoring polynomials with rational coefficients. Math Ann, 1982, 261: 515–534
21 Chen Y, Nguyen P Q. BKZ 2.0: better lattice security estimates. In: Advances in Cryptology–ASIACRYPT 2011. Berlin: Springer, 2011. 7073: 1–20
22 Schnorr C P, Euchner M. Lattice basis reduction: improved practical algorithms and solving subset sum problems. Math Programm, 1994, 66: 181–199
23 Aono Y, Nguyen P Q. Random sampling revisited: lattice enumeration with discrete pruning. In: Advances in Cryptology–EUROCRYPT 2017. Cham: Springer, 2017. 10211: 65–102
24 Aono Y, Wang Y, Hayashi T, et al. Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. In: Advances in Cryptology–EUROCRYPT 2016. Berlin: Springer, 2016. 9665: 789–819
25 Zheng Z X, Wang X Y, Xu G W, et al. Orthogonalized lattice enumeration for solving SVP. Sci China Inf Sci, 2018, 61: 032115
26 Micciancio D, Goldwasser S. Complexity of Lattice Problems: a Cryptographic Perspective. Norwell: Kluwer Academic Publishers, 2002. 14–22
27 Schnorr C P. A hierarchy of polynomial time lattice basis reduction algorithms. Theor Comput Sci, 1987, 53: 201–224
28 Schnorr C P, Hörner H. Attacking the Chor-Rivest cryptosystem by improved lattice reduction. In: Advances in Cryptology–EUROCRYPT'95. Berlin: Springer, 1995. 921: 1–12
29 Babai L. On Lovász' lattice reduction and the nearest lattice point problem. Combinatorica, 1986, 6: 1–13
30 Kannan R. Algorithmic geometry of numbers. Annu Rev Comput Sci, 1987, 2: 231–267
31 Shoup V. Number theory C++ library (NTL) vesion 6.0.0. http://www.shoup.net/ntl/
32 Kannan R. Minkowski's convex body theorem and integer programming. Math Oper Res, 1987, 12: 415–440

Appendix A Proof of Lemma 4

Since $\|\boldsymbol{r}\|_\infty < \kappa\eta$, we have

$$
\begin{aligned}
&|r_i| = |ne_i + 2^{l'}y + \overline{x}_i t_{i,1} - t_{i,2}| < \kappa\eta < 1, \quad \text{for } 1 \leqslant i \leqslant m,\\
&|r_{m+1}| = \left|\frac{2^{l'}\eta}{n}y\right| < \kappa\eta,\\
&|r_{m+2i}| = \left|\frac{\eta}{2^l}t_{i,1}\right| < \kappa\eta < 1, \quad \text{for } 1 \leqslant i \leqslant m,\\
&|r_{m+2i+1}| = \left|\frac{\eta}{2^{l'}}t_{i,2}\right| < \kappa\eta < 1, \quad \text{for } 1 \leqslant i \leqslant m,
\end{aligned}
$$

which implies

$$
\begin{aligned}
&2^{l'}y + \overline{x}_i t_{i,1} - t_{i,2} = 0 \bmod n, \quad \text{for } 1 \leqslant i \leqslant m,\\
&|y| < \frac{\kappa n}{2^{l'}},\\
&|t_{i,1}| < \kappa 2^l, \quad |t_{i,2}| < \kappa 2^{l'}, \quad \text{for } 1 \leqslant i \leqslant m.
\end{aligned}
$$

Assume the following relations for $1 \leqslant i \leqslant m$,

$$
\begin{aligned}
&2^{l'}y = \kappa_y n + 2^{l'}y_1 + y_0 \text{ with } |\kappa_y| < \kappa,\ 0 \leqslant y_1 < n/2^{l'} \text{ and } 0 \leqslant y_0 < n,\\
&t_{i,2} = \kappa_i 2^{l'} + t'_{i,2} \text{ with } 0 \leqslant t'_{i,2} < 2^{l'} \text{ and } |\kappa_i| < \kappa,\\
&t'_{i,1} = t_{i,1} + \overline{x}_i{}^{-1}(y_0 - \kappa_i 2^{l'}) \bmod n,\\
&\overline{x}_i \cdot \overline{x}_i{}^{-1} = 1 + \kappa'_i n,
\end{aligned}
$$

$$e'_i = e_i + \kappa_y + \kappa'_i(y_0 - \kappa_i 2^{l'}).$$

Then the vector $\boldsymbol{r'} = \boldsymbol{z'B}$ is also a lattice point with coordinates vector $\boldsymbol{z} = (e'_1, \ldots, e'_m, y_1, t'_{1,1}, t'_{1,2}, \ldots, t'_{i,1}, t'_{i,2}, \ldots, t'_{m,1}, t'_{m,2})$, which satisfies

$$\begin{aligned}
|r'_i| &= |ne'_i + 2^{l'} y_1 + \overline{x}_i t'_{i,1} - t'_{i,2}| \\
&= |ne'_i + 2^{l'} y - \kappa_y n - y_0 + \overline{x}_i(t_{i,1} + \overline{x}_i{}^{-1}(y_0 - \kappa_i 2^{l'})) + \kappa_i n 2^{l'} - t_{i,2}| \\
&= |n(e'_i - \kappa_y - \kappa'_i(y_0 - \kappa_i 2^{l'})) + 2^l y + \overline{x}_i t_{i,1} - t_{i,2}| \\
&= |ne_i + 2^{l'} y + \overline{x}_i t_{i,1} - t_{i,2}| \\
&= |r_i| < \kappa\eta < 1, \quad \text{for } 1 \leqslant i \leqslant m.
\end{aligned}$$

This implies that $ne'_i + 2^{l'} y_1 + \overline{x}_i t'_{i,1} - t'_{i,2} = 0$. Therefore, we obtain

$$2^{l'} y_1 + \overline{x}_i t'_{i,1} - t'_{i,2} \equiv 0 \bmod n, \quad \text{for } 1 \leqslant i \leqslant m.$$

For some fixed y, consider the event E defined as $2^{l'} y_1 - t'_{i,2} \neq 0 \bmod n$ for all i. We first prove the upper bound for the probability that E happens. For any $1 \leqslant i \leqslant m$, we have

$$\overline{x}_i{}^{-1} = t'_{i,1}(2^{l'} y_1 - t'_{i,2})^{-1} \bmod n. \tag{A1}$$

There exist at most $(2\kappa 2^l - 1) \cdot (2\kappa 2^{l'} - 1)$ number of possible tuples $(t_{i,1}, t_{i,2})$ which lead to a non-zero $\overline{x}_i{}^{-1}$. Thus the probability that Eq. (A1) has a non-zero solution on $(t_{i,1}, t_{i,2})$ is less than

$$P_i(y) = \frac{(2\kappa 2^l - 1) \cdot (2\kappa 2^{l'} - 1)}{n} \leqslant \frac{\kappa^2 2^{l+l'+2}}{n}.$$

Because all $\{\overline{x}_i\}_{i=1}^m$ are independent, the probability that E happens under the condition of fixed y is less than

$$\prod_{i=1}^{m} P_i(y) \leqslant \frac{\kappa^{2m} 2^{(l+l'+2)m}}{n^m}.$$

Denote P_E as the probability that E happens. With the bound for y, we obtain

$$\begin{aligned}
P_E &\leqslant \frac{\kappa n}{2^{l'}} \cdot \frac{\kappa^{2m} 2^{(l+l'+2)m}}{n^m} \\
&= \frac{\kappa^{2m+1} 2^{(l+l'+2)m-l'}}{n^{m-1}}.
\end{aligned}$$

Consequently, there exists some $w \in [1, m]$ such that

$$2^{l'} y_1 - t'_{w,2} = 0 \bmod n$$

holds with probability $1 - P_E$.

Since $t'_{w,2}$ and y_1 are bounded by $2^{l'}$ and $n/2^{l'}$ respectively, we get

$$2^{l'} y_1 = t'_{w,2} = 0,$$

which implies $y_1 = 0$ and this completes the proof.