

Research Article

Dimitrios Poulakis*

New lattice attacks on DSA schemes

DOI: 10.1515/jmc-2014-0027

Received July 23, 2014; accepted April 21, 2016

Abstract: We prove that a system of linear congruences of a particular form has at most a unique solution below a certain bound which can be computed efficiently. Using this result, we develop attacks against the DSA schemes which, under some assumptions, can provide the secret key in the case where one or several signed messages are available.

Keywords: Public key cryptography, digital signature algorithm, elliptic curve digital signature algorithm, closest vector problem, discrete logarithm

MSC 2010: 94A60, 11T71, 11Y16

Communicated by: Spyros Magliveras

1 Introduction

In August 1991, the U.S. government's National Institute of Standards and Technology (NIST) proposed an algorithm for digital signatures. The algorithm is known as DSA, for Digital Signature Algorithm [11, 14, 20]. It is an efficient variant of the ElGamal digital signature scheme [6] intended for use in electronic mail, electronic funds transfer, electronic data interchange, software distribution, data storage, and other applications which require data integrity assurance and data authentication. In 1998, an elliptic curve analogue called Elliptic Curve Digital Signature Algorithm (ECDSA) was proposed and standardized [10–12]. In the next subsection we shall recall the outlines of DSA and ECDSA.

1.1 The DSA schemes

First, for DSA, the signer chooses a prime p of size between 512 and 1024 bits with increments of 64, q is a prime of size 160 with $q | p - 1$, and g is a generator of the unique order q subgroup G of \mathbb{Z}_p^* . The signer chooses $a \in \{1, \dots, q - 1\}$ and computes $A = g^a \bmod p$. His public key is (p, q, g, A) and his private key a . Furthermore, the signer chooses a publicly known hash function h mapping messages to $\{0, \dots, q - 1\}$. To sign a message m , he chooses a random number $k \in \{1, \dots, q - 1\}$ which is the ephemeral key, computes

$$r = (g^k \bmod p) \bmod q \quad \text{and} \quad s = k^{-1}(h(m) + ar) \bmod q.$$

The signature of m is the pair (r, s) . The signature is valid if and only if the following equality holds:

$$r = ((g^{s^{-1}h(m)} \bmod q) A^{s^{-1}r} \bmod q) \bmod p) \bmod q.$$

The ECDSA uses an elliptic curve E over \mathbb{F}_p and a point $P \in E(\mathbb{F}_p)$ with order a prime q of size around 160 bits. The signer selects $a \in \{1, \dots, q - 1\}$ and computes $Q = aP$. Its public key is (p, E, P, q, Q) and his

*Corresponding author: Dimitrios Poulakis: Department of Mathematics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece, e-mail: poulakis@math.auth.gr

private key a . To sign a message m having hash value $h(m) \in \{0, \dots, q-1\}$, he selects a random number $k \in \{1, \dots, q-1\}$ which is the ephemeral key and computes $kP = (x, y)$ (where x and y are regarded as integer between 0 and $p-1$). Next, he computes

$$r = x \bmod q \quad \text{and} \quad s = k^{-1}(h(m) + ar) \bmod q.$$

The signature of m is the pair (r, s) . For the verification of the signature one computes

$$u_1 = s^{-1}h(m) \bmod q, \quad u_2 = s^{-1}r \bmod q, \quad u_1P + u_2Q = (x_0, y_0).$$

One accepts the signature if and only if $r = x_0 \bmod q$.

The security of DSA (respectively ECDSA) relies on the hardness of the discrete logarithm problem in prime fields and their subgroups (respectively in the group of elliptic curve). Thus, the parameters of the two systems were chosen in such a way that the computation of discrete logarithms is computationally infeasible, and so the secret key a and the ephemeral key k are well protected.

1.2 Related works

The use of lattices and the LLL reduction method [13] is an efficient tool for attacking a variety of cryptographic schemes. Attacks to DSA schemes using lattice reduction techniques and the equality

$$s = k^{-1}(h(m) + ar) \bmod q$$

are given in [1, 2, 5, 7, 9, 16–18].

In [1], Bellare et al. show that if random numbers for DSA are generated using a linear congruential pseudorandom number generator (LCG), then the combination of the DSA “signature equations” with the LCG generation equations leads to a system of equations which provide the secret key. Babai’s closest vector approximation algorithm is used to solve such a system. This algorithm only returns approximations to the closest vector which are not very good.

In [9], Howgrave-Graham and Smart propose several heuristic attacks to recover the secret key under the hypothesis that for a reasonable number of signatures, a small fraction of the corresponding nonce k is revealed. As in [1], the attacks are based on the LLL-based Babai CVP approximation algorithm. They use several heuristic assumptions which do not allow precise statements on the algorithm’s theoretical behavior.

In [16], Nguyen and Shparlinski give the first provable polynomial-time attack against DSA under the hypotheses that the size of q is not too small compared with p , and the probability of collisions for the hash function h is not too large compared with $1/q$. More precisely, if for a certain (polynomially bounded) number of random messages m and random nonces k , about $\log^{1/2} q$ least significant bits of k are known, then in polynomial time one can recover the signer’s secret key a . This attack is adapted to the case of ECDSA [17].

In [2], Blake and Garfalakis, using one message, compute with the LLL reduction method two short vectors of a three-dimensional lattice and, in case where the second shortest vector is sufficiently short, they deduce two lines which intersect in (a, k) , provided that a and k are sufficiently small. If two messages are available, one has a linear congruence relating the corresponding ephemeral keys and the same attack is applicable.

The attack presented by the author in [18] uses also one message. The algorithm LLL and two algorithms for the computation of the integral points of two classes of conics are combined for the computation of the secret key provided that at least the elements of one of the sets $\{a, k^{-1} \bmod q\}$, $\{k, a^{-1} \bmod q\}$ and $\{a^{-1} \bmod q, k^{-1} \bmod q\}$ are sufficiently small. As in the previous attack, if two messages are available, we can apply these attacks to the congruence relating the two ephemeral keys.

In [5], the author together with Draziotis use a two-dimensional lattice L which is defined by a signed message. Lagrange’s lattice reduction algorithm provides a basis of L formed by two successive minima. Using this basis, we construct two linear polynomials $f_i(x, y)$ such that (a, k) is the intersection point of two straight lines of the form $f_i(x, y) = c_i q$, where $c_i \in \mathbb{Z}$ ($i = 1, 2$). If a and k are sufficiently small, then both c_i belong

to a small set and so we can compute the secret key a in polynomial time. Similar attacks hold for the pairs $(k^{-1} \bmod q, k^{-1}a \bmod q)$ and $(a^{-1} \bmod q, a^{-1}k \bmod q)$. If we have two signed messages, then we can apply the same attacks to the equation relating the two ephemeral keys.

Finally, in [7], Faugère et al. assume that we only know that there are equalities between δ bits of the unknown ephemeral keys used to sign some messages. It is shown that this implicit information should be extracted by constructing a lattice which contains a very short vector such that its components yield the secret key. When the ephemeral keys share enough bits δ , this vector is small enough and so can be computed by the LLL lattice reduction algorithm. Furthermore, the Gaussian heuristic is used to find a condition on the number of shared bits δ in function of the number of messages for this vector to be the shortest of the lattice.

1.3 Our contribution

In this paper, we also use the equality $s = k^{-1}(h(m) + ar) \bmod q$ and we develop new attacks on the DSA schemes. More precisely, we first compute a lower bound for the size of vectors of a particular class of lattices. Using this result and the Micciancio–Voulgaris theorem [15] on the closest vector problem, we prove that a system of n linear congruences, where $n \leq \log \log q - 1$, of a particular form has at most a unique solution of size smaller than $q^{n/(n+1)}/16$ which can be computed in polynomial time. Thus, using only a signed message, we can construct such a system of linear congruences, which, in the case where it satisfies some condition, provides us the secret key in polynomial time. If we have two signed messages, then we can perform the attack to the linear congruence relating the corresponding ephemeral keys. Further, we give some variants of this attacks using the modular inverses of the keys.

When more than one message is available, it is more possible to construct the linear system with the required properties. If we possess a small fixed number of signed messages, say ≤ 20 , the time complexity of the attack remains polynomial. On the other hand, if we use the maximum number of $\log \log q - 1$ messages, then our attack has subexponential running time.

Our attack is rigorous while the attacks in [7, 9] use heuristic assumptions. Furthermore, it is independent from the pseudorandom number generator which is used for the generation of the random numbers. The attacks in [16, 17] use polynomially bounded number of random messages and in [7] some messages which are signed with ephemeral keys sharing enough bits, while our attacks can be performed with only one message. The attacks in [5, 18] use also one message but they compute the secret key in polynomial time in the case where at least one of the keys or its modular inverse is at most $q^{1/2} + O(\log q)$. This bound in our attack is $q^{n/(n+1)}/16$. On the other hand, a disadvantage of our attacks is that some hypotheses depending on the available signed messages often are not satisfied with the result that our attack cannot be applied to. Finally, note that when our attacks are applicable, we can compute a secret key of full size in the case where its inverse is less than $q^{n/(n+1)}/4$ as the example in Section 5 shows. Thus, in order for the DSA schemes to be protected from our attacks, the secret, the ephemeral keys and their modular inverses must have the same size as q .

1.4 The structure of the paper

The paper is organized as follows. In Section 2, we recall some basic results about lattices as well as the Micciancio–Voulgaris theorem on the closest vector problem. Moreover, we prove a key lemma for later use. In Section 3, we prove our theorem on the solutions of a system of linear congruences. Our attacks based on it are presented in Section 4. An example is given in Section 5 and finally Section 6 concludes the paper.

2 Auxiliary results

Let $B = \{\mathbf{b}_1, \dots, \mathbf{b}_n\} \subset \mathbb{Z}^n$ be a basis of \mathbb{R}^n . An n -dimensional lattice spanned by B is the set

$$\mathcal{L} = \{z_1 \mathbf{b}_1 + \dots + z_n \mathbf{b}_n : z_1, \dots, z_n \in \mathbb{Z}\}.$$

The *Euclidean norm* of a vector $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{R}^n$ is defined to be the quantity

$$\|\mathbf{v}\| = (v_1^2 + \dots + v_n^2)^{1/2}.$$

Throughout the paper we state \log for the natural logarithm and \log_2 for the logarithm with base 2.

Lemma 2.1. *Let $q > 2$ be a prime and let n, A_i ($i = 1, \dots, n$) be integers satisfying*

$$0 < n \leq \log \log q - 1, \quad 2^{i-1} q^{i/(n+1)} < A_i < 2^i q^{i/(n+1)}.$$

We denote by \mathcal{L} the lattice spanned by the rows of the square matrix

$$J = \begin{pmatrix} -1 & A_1 & A_2 & \dots & A_n \\ 0 & q & 0 & \dots & 0 \\ 0 & 0 & q & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & q \end{pmatrix}.$$

Then for every nonzero $\mathbf{v} \in \mathcal{L}$ we have

$$\|\mathbf{v}\| > \frac{q^{n/(n+1)}}{8}.$$

Proof. First, we show that $A_n < q$. We have $2^n q^{n/(n+1)} < q$ if and only if $2^n < q^{1/(n+1)}$ which is equivalent to $n(n+1) < \log_2 q$. Further,

$$n(n+1) \leq (\log \log q - 1)(\log \log q) \leq (\sqrt{\log q} - 1)\sqrt{\log q} < \log_2 q.$$

It follows that $A_n < q$.

Suppose next that there are integers x_0, \dots, x_n such that

$$\mathbf{v} = (-x_0, x_0 A_1 + x_1 q, \dots, x_0 A_n + x_n q)$$

is a nonzero vector of \mathcal{L} satisfying

$$\|\mathbf{v}\| \leq \frac{q^{n/(n+1)}}{8}.$$

Then we have

$$\max\{|x_0|, |x_0 A_1 + x_1 q|, \dots, |x_0 A_n + x_n q|\} < \frac{q^{n/(n+1)}}{8}.$$

If $x_0 = 0$, then there is $i \in \{1, \dots, n\}$ such that $x_i \neq 0$ and so we get $\|\mathbf{v}\| > q$ which is a contradiction. Thus $x_0 \neq 0$. If $x_0 A_j + x_j q = 0$, then $x_j \neq 0$. Thus $q \mid x_0 A_j$ and so we get either $q \mid x_0$ or $q \mid A_j$. On the other hand we have $0 < |x_0| < q$ and $0 < A_j < A_n < q$ which is a contradiction. Hence $x_0 A_j + x_j q \neq 0$ ($j = 1, \dots, n$).

Let $i \in \{1, \dots, n\}$ such that

$$\frac{q^{(n-i)/(n+1)}}{2^{i+2}} < |x_0| < \frac{q^{(n-i+1)/(n+1)}}{2^{i+1}}.$$

If $x_j \neq 0$ with $1 \leq j \leq i$, then we have

$$\frac{q^{n/(n+1)}}{8} > |x_0 A_j + x_j q| \geq q - |x_0| A_j > q - \frac{q^{(n-i+1)/(n+1)}}{2^{i+1}} A_i > \frac{q}{2},$$

whence we get $1 > 4q^{1/(n+1)}$ which is a contradiction. Hence $x_j = 0$ ($j = 1, \dots, i$). Then we have

$$\|\mathbf{v}\| > |x_0 A_i| > \frac{q^{(n-i)/(n+1)}}{2^{i+2}} 2^{i-1} q^{i/(n+1)} = \frac{q^{n/(n+1)}}{8}.$$

which is a contradiction. The result follows. \square

Theorem 2.2. Let \mathcal{L} be an n -dimensional lattice and $\mathbf{y} \in \mathbb{R}^n$. Then there is a deterministic algorithm that computes $\mathbf{v} \in \mathcal{L}$ such that for every $\mathbf{t} \in \mathcal{L}$ we have

$$\|\mathbf{v} - \mathbf{y}\| \leq \|\mathbf{t} - \mathbf{y}\|$$

in time $2^{2n+o(n)}$.

Proof. See [15]. □

3 A system of linear congruences

Our attacks are based on the following theorem.

Theorem 3.1. Let $q > 2$ be a prime and let n, A_i, B_i ($i = 1, \dots, n$) be positive integers such that $n \leq \log \log q - 1$ and $2^{i-1}q^{i/(n+1)} < A_i < 2^i q^{i/(n+1)}$. Then the system of congruences

$$y_i + A_i x + B_i \equiv 0 \pmod{q} \quad (i = 1, \dots, n)$$

has at most one solution $\mathbf{v} = (x, y_1, \dots, y_n) \in \{0, \dots, q-1\}^{n+1}$ having

$$\|\mathbf{v}\| < \frac{q^{n/(n+1)}}{16}.$$

If such a solution exists, then the algorithm of Theorem 2.2 applied to the vector $(0, B_1, \dots, B_n)$ and the lattice of Lemma 2.1 gives a vector \mathbf{w} whose first coordinate is x . The time complexity of computation of x is $O((\log q)^2)$.

Proof. Let $\mathbf{v} = (x, y_1, \dots, y_n)$ be a solution of the system with

$$\|\mathbf{v}\| < \frac{q^{n/(n+1)}}{16}.$$

Consider the lattice \mathcal{L} spanned by the rows of the square matrix

$$J = \begin{pmatrix} -1 & A_1 & A_2 & \dots & A_n \\ 0 & q & 0 & \dots & 0 \\ 0 & 0 & q & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & q \end{pmatrix}$$

and set $\mathbf{b} = (0, B_1, \dots, B_n)$. Since $y_i + A_i x + B_i \equiv 0 \pmod{q}$, there is $z_i \in \mathbb{Z}$ such that $-xA_i + z_i q = y_i + B_i$ ($i = 1, \dots, n$). Then the vector $\mathbf{u} = (x, -xA_1 + z_1 q, \dots, -xA_n + z_n q)$ belongs to \mathcal{L} and we have

$$\|\mathbf{u} - \mathbf{b}\| = \|(x, y_1, \dots, y_n)\| = \|\mathbf{v}\| < \frac{q^{n/(n+1)}}{16}.$$

On the other hand, by Theorem 2.2 we can compute $\mathbf{w} \in \mathcal{L}$ such that

$$\|\mathbf{w} - \mathbf{b}\| \leq \|\mathbf{u} - \mathbf{b}\| < \frac{q^{n/(n+1)}}{16}.$$

The time complexity of the computation is $2^{2n+o(n)} = O((\log q)^2)$. Thus, we have

$$\|\mathbf{u} - \mathbf{w}\| \leq \|\mathbf{u} - \mathbf{b}\| + \|\mathbf{w} - \mathbf{b}\| < \frac{q^{n/(n+1)}}{8}.$$

Since $\mathbf{u} - \mathbf{w} \in \mathcal{L}$, Lemma 2.1 implies $\mathbf{u} = \mathbf{w}$. Thus, if $\mathbf{w} = (w_0, \dots, w_n)$, then we get $x = w_0$. □

4 Our attacks

In this section we describe our attacks. Let n be a positive integer $\leq \log \log q - 1$. Suppose we have $t \leq n$ signed messages m_j ($j = 1, \dots, t$) and their signatures (r_j, s_j) , respectively, with DSA (resp. ECDSA). Then there are $k_j \in \{1, \dots, q-1\}$ such that $r_j = (g^{k_j} \bmod p) \bmod q$ (resp. $r_j = x_j \bmod q$ and $k_j P = (x_j, y_j)$) and $s_j = k_j^{-1}(h(m_j) + ar_j) \bmod q$. It follows that

$$k_j + C_j a + D_j \equiv 0 \pmod{q} \quad (j = 1, \dots, t),$$

where $C_j = -r_j s_j^{-1} \bmod q$ and $D_j = -s_j^{-1} h(m_j) \bmod q$.

We give below an algorithm based on Theorem 3.1 which computes a under some assumptions.

DSA-ATTACK-1. *Input:* (m_j, r_j, s_j) ($j = 1, \dots, t$).

1. Compute $C_j = -r_j s_j^{-1} \bmod q$ and $D_j = -s_j^{-1} h(m_j) \bmod q$.
2. Select integers A_i ($i = 1, \dots, n$) with $2^{i-1} q^{i/(n+1)} < A_i < 2^i q^{i/(n+1)}$. (If $2^{i-1} q^{i/(n+1)} < C_i < 2^i q^{i/(n+1)}$, then we can take $A_i = C_i$.)
3. Compute $B_{ij} = A_i D_j C_j^{-1} \bmod q$ ($i = 1, \dots, n, j = 1, \dots, t$).
4. Denote by M the set of maps $\mu : \{1, \dots, n\} \rightarrow \{1, \dots, t\}$. For every $\mu \in M$, compute (as it is explained in Theorem 3.1) the coefficient x_μ of the solution $\mathbf{v}_\mu = (x_\mu, y_{1,\mu(1)}, \dots, y_{n,\mu(n)})$ of the system

$$y_i + A_i x + B_{i,\mu(i)} \equiv 0 \pmod{q} \quad (i = 1, \dots, t)$$

with $\|\mathbf{v}_\mu\| < q^{n/(n+1)}/16$.

5. For every $\mu \in M$ check if x_μ is the private key a .

The performance of the algorithm is given in the following proposition.

Proposition 4.1. *Put $k_{ij} = k_j A_i C_j^{-1} \bmod q$ ($i = 1, \dots, n, j = 1, \dots, t$). Suppose there is $\mu \in M$ such that*

$$\|(a, k_{1\mu(1)}, \dots, k_{n\mu(n)})\| < \frac{q^{n/(n+1)}}{16}.$$

Then the algorithm DSA-ATTACK-1 computes a . Its time complexity for the case of DSA is $O((\log p)^2 (\log q)^{1+\log t})$ bit operations and for ECDSA $O((\log q)^2 ((\log q)^{\log t} + (\log \log q)^4))$ bit operations and $O((\log q)^{1+\log t})$ elliptic curves operations.

Proof. Steps 1, 2, 3 and 4 build for every $\mu \in M$ the system of congruences

$$y_i + A_i x + B_{i,\mu(i)} \equiv 0 \pmod{q} \quad (i = 1, \dots, t).$$

Step 1 needs $O((\log q)^2 t)$ bit operations. The computation of each q^i needs time $O(i^2 (\log q)^2)$. The required time for the computation of $\lfloor q^{i/(n+1)} \rfloor$ is $O(i^2 (\log q)^2)$ (see [4]) and for $2^i \lfloor q^{i/(n+1)} \rfloor$ is $O(i^3 (\log q)^2)$. Thus, we can select A_i such that

$$2^{i-1} \lfloor q^{i/(n+1)} \rfloor + 2^{i-1} \leq A_i \leq 2^i \lfloor q^{i/(n+1)} \rfloor.$$

So, the time complexity of Step 2 is

$$O\left(\sum_{i=1}^n i^3 (\log q)^2\right) = O((\log q)^2 n^4) = O((\log q)^2 (\log \log q)^4).$$

Step 3 needs $O((\log q)^2 (\log \log q)^2)$ bit operations. Step 4 requires t^n times the application of the algorithm of Theorem 2.2 and so its time complexity is $O((\log q)^{2+\log t})$. Therefore, the time complexity of the construction of all the systems and the computation of x_μ is

$$O((\log q)^2 ((\log q)^{\log t} + (\log \log q)^4)).$$

The complexity of Step 5 is $O((\log p)^2 (\log q)^{1+t})$ bit operations in the case of DSA and $O((\log q)^{1+t})$ elliptic curves operations in the case of ECDSA. Therefore, for DSA, the time complexity of the procedure is

$O((\log p)^2(\log q)^{1+t})$ bit operations and for ECDSA $O((\log q)^2((\log q)^{\log t} + (\log \log q)^4))$ bit operations and $O((\log q)^{1+t})$ elliptic curves operations.

For every $\mu \in M$, the vector $(a, k_{1,\mu(1)}, \dots, k_{n,\mu(n)})$ is a solution of the system of congruences

$$y_i + A_i x + B_{i,\mu(i)} \equiv 0 \pmod{q} \quad (i = 1, \dots, t).$$

Thus, in the case where there is $\mu \in M$ satisfying

$$\|(a, k_{1\mu(1)}, \dots, k_{n\mu(n)})\| < \frac{q^{n/(n+1)}}{16},$$

Theorem 3.1 implies that a is among the elements x_μ and so it can be computed using DSA-ATTACK-1. \square

Multiplying the congruence

$$k_j + C_j a + D_j \equiv 0 \pmod{q} \quad (j = 1, \dots, t)$$

by a^{-1} , we get

$$k_j a^{-1} + C_j + D_j a^{-1} \equiv 0 \pmod{q} \quad (j = 1, \dots, t).$$

Thus, replacing (C_j, D_j) by (D_j, C_j) and a by a^{-1} , we obtain a variant of DSA-ATTACK-1 which under some assumption provides us $a^{-1} \pmod{q}$ and so a . We call it DSA-ATTACK-2. Thus, we have the following result.

Proposition 4.2. Put $k_{ij} = k_j a^{-1} A_i D_j^{-1} \pmod{q}$ ($i = 1, \dots, n, j = 1, \dots, t$). Suppose there is $\mu \in M$ such that

$$\|(a^{-1} \pmod{q}, k_{1\mu(1)}, \dots, k_{n\mu(n)})\| < \frac{q^{n/(n+1)}}{16}.$$

Then the algorithm DSA-ATTACK-2 computes a . Its time complexity for the case of DSA is $O((\log p)^2(\log q)^{1+\log t})$ bit operations and for ECDSA $O((\log q)^2((\log q)^{\log t} + (\log \log q)^4))$ bit operations and $O((\log q)^{1+\log t})$ elliptic curves operations.

Suppose now that $t \geq 2$. So we can eliminate a among the congruences

$$k_j + C_j a + D_j \equiv 0 \pmod{q} \quad (j = 1, \dots, t)$$

and we deduce the congruences

$$k_j + \tilde{C}_j k_t + \tilde{D}_j \equiv 0 \pmod{q} \quad (j = 1, \dots, t-1),$$

where $\tilde{C}_j = -C_j C_t^{-1} \pmod{q}$ and $\tilde{D}_j = -C_j C_t^{-1} D_t + D_j \pmod{q}$. Replacing in DSA-ATTACK-1 (C_j, D_j) by $(\tilde{C}_j, \tilde{D}_j)$, we have another variant of DSA-ATTACK-1 which we call DSA-ATTACK-3 and which under some assumption provides us a . Hence, we have the following result.

Proposition 4.3. Put $k_{ij} = k_j A_i \tilde{C}_j^{-1} \pmod{q}$ ($i = 1, \dots, n, j = 1, \dots, t-1$). Suppose there is $\mu \in M$ such that

$$\|(k_t, k_{1\mu(1)}, \dots, k_{n\mu(n)})\| < \frac{q^{n/(n+1)}}{16}.$$

Then the algorithm DSA-ATTACK-3 computes a . Its time complexity for the case of DSA is $O((\log p)^2(\log q)^{1+\log t})$ bit operations and for ECDSA $O((\log q)^2((\log q)^{\log t} + (\log \log q)^4))$ bit operations and $O((\log q)^{1+\log t})$ elliptic curves operations.

Furthermore, multiplying the congruences

$$k_j + \tilde{C}_j k_t + \tilde{D}_j \equiv 0 \pmod{q} \quad (j = 1, \dots, t-1)$$

by k_t^{-1} , we obtain

$$k_j k_t^{-1} + \tilde{C}_j + \tilde{D}_j k_t^{-1} \equiv 0 \pmod{q} \quad (j = 1, \dots, t-1).$$

So, replacing (C_j, D_j) in DSA-ATTACK-2 by $(\tilde{C}_j, \tilde{D}_j)$, we have another attack which we call DSA-ATTACK-4.

Proposition 4.4. Put $k_{ij} = k_j k_t^{-1} A_i \tilde{D}_j^{-1} \bmod q$ ($i = 1, \dots, n, j = 1, \dots, t$). Suppose there is $\mu \in M$ such that

$$\|(k_t^{-1} \bmod q, k_{1\mu(1)}, \dots, k_{n\mu(n)})\| < \frac{q^{n/(n+1)}}{16}.$$

Then the algorithm DSA-ATTACK-4 computes a . Its time complexity for the case of DSA is $O((\log p)^2 (\log q)^{1+\log t})$ bit operations and for ECDSA $O((\log q)^2 ((\log q)^{\log t} + (\log \log q)^4))$ bit operations and $O((\log q)^{1+\log t})$ elliptic curves operations.

Remark 4.5. If t is a very small fixed number, say $t < 20$, then the time complexity of our attacks is polynomial.

Remark 4.6. Note that the greater the number of available messages, the more likely is the construction of a linear system with the desired properties.

5 An example

We consider the elliptic curve E given in [3, p.182, Example 3] defined over the finite field \mathbb{F}_p , where $p = 2^{160} + 7$ is a prime, by the equation

$$y^2 = x^3 + 10x + 1343632762150092499701637438970764818528075565078.$$

The number of points of $E(\mathbb{F}_p)$ is the 160-bit prime

$$q = 1461501637330902918203683518218126812711137002561.$$

Consider the point $P = (x(P), y(P))$ of $E(\mathbb{F}_p)$ of order q , where

$$\begin{aligned} x(P) &= 858713481053070278779168032920613680360047535271, \\ y(P) &= 364938321350392265038182051503279726748224184066. \end{aligned}$$

We take as private key the 160-bit integer

$$a = 874984668032211733311386841306673749333236586178.$$

The public key is $Q = aP = (x(Q), y(Q))$ where

$$\begin{aligned} x(Q) &= 597162246892872056034315330452950636324741691536, \\ y(Q) &= 1181877329208353060566969266758924757549684357390. \end{aligned}$$

Let m_1, m_2 and m_3 be three messages with hash values

$$\begin{aligned} h(m_1) &= 1238458437157734227527825004718505271235024916418, \\ h(m_2) &= 1028653949698644928576637572550961266718086213222, \\ h(m_3) &= 1359253753908721564345086919389145449479510713328. \end{aligned}$$

Suppose that the following ephemeral keys have been used respectively for the generation of the signatures of the three messages:

$$\begin{aligned} k_1 &= 466080543322889688835467115835518398826523750031, \\ k_2 &= 730750818665451459101842416358141509827966271589, \\ k_3 &= 730750818665451459101842416358141509827966279681. \end{aligned}$$

The size of k_1 is 158 bits and the size of k_2 and k_3 is 159 bits. We have the points $R_i = k_i P = (x(R_i), y(R_i))$ ($i = 1, 2, 3$), where

$$\begin{aligned} x(R_1) &= 1254157729089443995418123832523808277031313949462, \\ y(R_1) &= 23109942117176529567525517253616649087109941040, \\ x(R_2) &= 725144377910246885534616706756699404195507663231, \\ y(R_2) &= 724834174614588160856240480005855379930897712013, \\ x(R_3) &= 250593598147858114836913138265564915457464710851, \\ y(R_3) &= 63119281333557571230379851501639067328261656282. \end{aligned}$$

The signature of m_i is (r_i, s_i) where $s_i = k_i^{-1}(h(m_i) + ar_i) \bmod q$ and $r_i = x(R_i)$ ($i = 1, 2, 3$). We have

$$\begin{aligned} s_1 &= 1363805341335356352807650823690154552653914451119, \\ s_2 &= 1286644068312084224467989193436769265471767284571, \\ s_3 &= 1357235540051781293143720232752751840677247754090. \end{aligned}$$

First, we remark that

$$a^{-1} \bmod q = 5070602400912917605986812821509 < 2^{103}.$$

Thus, we shall apply DSA-ATTACK-2 with $t = n = 3$. The couple $(a^{-1} \bmod q, k_j a^{-1} \bmod q)$ is a solution of the congruence

$$y + D_i x + C_i \equiv 0 \pmod{q} \quad (i = 1, 2, 3),$$

where

$$\begin{aligned} C_1 &= 1461501463106331049611349884018124821212302099515, \\ D_1 &= 34359738369, \\ C_2 &= 856585227192969567381714973407499157966149117422, \\ D_2 &= 1389773565760524781352174297091678638955836274432, \\ C_3 &= 25289181258142448854230843836548288088082171610, \\ D_3 &= 494393186466616365369065630169592100192862982492. \end{aligned}$$

We have

$$\begin{aligned} \lfloor q^{1/4} \rfloor &= 1099511627775, \\ \lfloor q^{1/2} \rfloor &= 1208925819614629174706175, \\ \lfloor q^{3/4} \rfloor &= 1329227995784915872903806163633513155. \end{aligned}$$

Next, we have to consider integers A_i ($i = 1, 2, 3$) satisfying

$$2^{i-1} q^{i/4} < A_i < 2^i q^{i/4} \quad (i = 1, 2, 3).$$

So, we take $A_1 = D_1$, $A_2 = 2^{81} + 1$ and $A_3 = 2^{122} + 23$.

Since we have

$$\begin{aligned} l_1 &= a^{-1} k_1 \bmod q < 2^{91}, \\ l_2 &= k_2 a^{-1} A_2 D_2^{-1} \bmod q < 2^{90}, \\ l_3 &= k_3 a^{-1} A_3 D_3^{-1} \bmod q < 2^{50}, \end{aligned}$$

we obtain

$$\|(a^{-1} \bmod q, l_1, l_2, l_3)\| < \frac{q^{3/4}}{16}.$$

Hence, the DSA-ATTACK-2 can provide us $a^{-1} \bmod q$ and so the secret key a .

6 Conclusion

In this paper we proposed some attacks on DSA schemes in the case where one or several signed messages are available. Using these messages, we built several systems of linear congruences which, in the case where their coefficients and solutions satisfy some inequalities, can provide us the secret key of the scheme. These attacks can also be applied to other signature schemes where the secret and the ephemeral keys are solutions of a linear congruence, such as the Schnorr signature, the Heyst–Pedersen signature, or GPS [8, 14, 19]. Note that in the case where the secret, the ephemeral keys and their modular inverses have the same size as q , our attacks cannot be applied.

References

- [1] M. Bellare, S. Goldwasser and D. Micciancio, “Pseudo-random” number generation within cryptographic algorithms: The DSS case, in: *Advances in Cryptology (CRYPTO '97)*, Lecture Notes in Comput. Sci. 1294, Springer, Berlin (1997), 277–291.
- [2] I. F. Blake and T. Garefalakis, On the security of the digital signature algorithm, *Des. Codes Cryptogr.* **26** (2002), no. 1–3, 87–96.
- [3] I. F. Blake, G. Seroussi and N. Smart, *Elliptic Curves in Cryptography*, Cambridge University Press, Cambridge, 2000.
- [4] R. Brent and P. Zimmerman, *Modern Computer Arithmetic*, Cambridge University Press, Cambridge, 2011.
- [5] K. Draziotis and D. Poulakis, Lattice attacks on DSA schemes based on Lagrange’s algorithm, in: *Algebraic Informatics (CAI 2013)*, Lecture Notes in Comput. Sci. 8080, Springer, Berlin (2013), 119–131.
- [6] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithm, *IEEE Trans. Inform. Theory* **31** (1985), 469–472.
- [7] J.-L. Faugère, C. Goyet and G. Renault, Attacking (EC)DSA given only an implicit hint, in: *Selected Area of Cryptography (SAC 2012)*, Lecture Notes in Comput. Sci. 7707, Springer, Berlin (2013), 252–274.
- [8] M. Girault, G. Poupard and J. Stern, Global Payment System (GPS): Un protocole de signature à la volée, Proceedings of Trusting Electronic Trade, 1999.
- [9] N. A. Howgrave-Graham and N. P. Smart, Lattice Attacks on Digital Signature Schemes, *Des. Codes Cryptogr.* **23** (2001), 283–290.
- [10] D. Johnson, A. J. Menezes and S. A. Vastone, The elliptic curve digital signature algorithm (ECDSA), *Int. J. Inf. Secur.* **1** (2001), 36–63.
- [11] N. Koblitz and A. J. Menezes, A survey of public-key cryptosystems, *SIAM Rev.* **46** (2004), no. 4, 599–634.
- [12] N. Koblitz, A. J. Menezes and S. A. Vastone, The state of elliptic curve cryptography, *Des. Codes Cryptogr.* **19** (2000), 173–193.
- [13] A. K. Lenstra, H. W. Lenstra, Jr. and L. Lovász, Factoring polynomials with rational coefficients, *Math. Ann.* **261** (1982), 513–534.
- [14] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, 1997.
- [15] D. Micciancio and P. Voulgaris, A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations, in: *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC '10)*, Association for Computing Machinery, New York (2010), 351–358.
- [16] P. Nguyen and I. E. Shparlinski, The insecurity of the digital signature algorithm with partially known nonces, *J. Cryptology* **15** (2002), 151–176.
- [17] P. Nguyen and I. E. Shparlinski, The insecurity of the elliptic curve digital signature algorithm with partially known nonces, *Des. Codes Cryptogr.* **30** (2003), 201–217.
- [18] D. Poulakis, Some Lattice Attacks on DSA and ECDSA, *Appl. Algebra Engrg. Comm. Comput.* **22** (2011), 347–358.
- [19] D. R. Stinson, *Cryptography, Theory and Practice*, 2nd ed., Chapman & Hall/CRC, Boca Raton, 2001.
- [20] National Institute of Standards and Technology (NIST), FIPS Publication 186: Digital Signature Standard, 1994.