Some Lattice Attacks on DSA and ECDSA

Dimitrios Poulakis
Department of Mathematics,
Aristotle University of Thessaloniki,
Thessaloniki 54124, Greece,
email:poulakis@math.auth.gr

November 10, 2010

Abstract

In this paper, using the LLL reduction method and computing the integral points of two classes of conics, we develop attacks on DSA and ECDSA in case where the secret and the ephemeral key and their modular inverse are quite small or quite large.

MSC 2010: 94A60, 11T71, 11Y16.
Keywords: Public Key Cryptography; Digital Signature Algorithm; Elliptic Curve Digital Signature Algorithm; Algorithm LLL; Discrete Logarithm; Diophantine Equations.

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 [13, 12, 10]. It is an efficient variant of the ElGamal digital signature scheme [4] 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 [5, 9, 10].

Let us recall the outlines of DSA and ECDSA. 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 $\mathbb{Z}_p^*$. Further, he chooses $a \in \{1,\ldots,q-1\}$ and computes $A = g^a \bmod p$. The public key of the signer is (p,q,g,A) and his private key a. Furthermore, the signer chooses a publicly known hash function h mapping messages to $\{0,\ldots,q-1\}$. To sign a message m, he chooses a random number $k \in \{1,\ldots,q-1\}$ 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 verification of the signature is performed by checking

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

The ECDSA uses an elliptic curve E over $\mathbb{Z}_p$ and a point $P \in E(\mathbb{Z}_p)$ 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 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

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

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

The assumption here is that the only way to forge signature is to recover either the secret key a, or the ephemeral key k (in this case is a simple matter to compute a). Thus, the parameters of the two systems were chosen in such a way that the computation of discrete logarithms is computationally infeasible, and so a or k is well protected.

The use of lattices and the so-called LLL reduction method [11] is a well established tool for attacking a variety of cryptosystems. Attacks to DSA and to ECDSA using lattice reduction techniques are given in [1], [8], [14], [15] and [2]. A common feature of these attacks is that take advantage of the form of equality $s = k^{-1}(h(m) + ar) \bmod q$. In [1] it was shown that one can recover the DSA secret key a, if the ephemeral key k is produced by Knuth's linear congruential generator with known parameters, or variants. In [8], an attack on DSA is described in case where for some number of different signatures a proportion of bits of each of the associated ephemeral keys are revealed. A polynomial-time attack on DSA which recover a is described in [14], in case where the size of q is not too small compared with p, the probability of collisions for the hash function is not too large compared to $1/q$ and for a polynomially bounded number of messages, about $\log_2^{1/2}(q)$ of the least significant bits of the ephemeral keys are known. The previous attack is adapted to the case of ECDSA [15]. Finally, in [2], under the assumption that the second shortest vector of the reduced lattice is sufficiently short, it is determined how large the keys a and k can be in order for them to be computed by considering only one signature.

In this paper, using the algorithm LLL and two algorithms for the computation of the integral points of two classes of conics, we present some new rigorous attacks on DSA and ECDSA which are based on the equality $s = k^{-1}(h(m) + ar) \bmod q$. Assuming that a signature is available and each number in at least one of the sets $\{a, k^{-1} \bmod q\}$, $\{k, a^{-1} \bmod q\}$ and $\{a^{-1} \bmod, k^{-1} \bmod q\}$ is smaller or larger that a certain explicit bound, we prove that the secret keys a and k can be revealed. Moreover, if two signatures with ephemeral keys k_1 and k_2 are available and each numbers in at least one of the sets $\{k_1, k_2^{-1} \bmod q\}$,

$\{k_2, k_1^{-1} \bmod q\}$ and $\{k_1^{-1} \bmod, k_2^{-1} \bmod q\}$ is smaller or larger that a certain explicit bound, then k_1, k_2 and so a can be computed.

In [16], we presented a version of the DSA which combines the intractability of the integer factorization problem and discrete logarithm problem, and it is at least as secure as DSA. It uses computations in the group $\mathbb{Z}_n^*$, where n is the product of two large primes which is part of the private key, and so the order of the underlying group is hidden. An immediate consequence of this fact is that the above mentioned attacks and the attacks described in this paper do not longer work.

The paper is organized as follows. In Section 2, some results on the LLL reduction method are recalled and two methods for the solution of Diophantine equations $bx + cy + dxy = 0$ and $b + cy + dxy = 0$ are given which are necessary for our attacks. Our attacks using one signed message are presented in Section 3. In Section 4, we deal with the attacks using two signed messages. Finally, Section 5 concludes the paper.

2 Auxiliary Results

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

$$L = \{z_1\mathbf{b}_1 + \cdots + z_n\mathbf{b}_n / \ z_1, \ldots, z_n \in \mathbb{Z}\}.$$

If $\mathbf{b}_i = (b_{i,1}, \ldots, b_{i,n})$ $(i = 1, \ldots, n)$, then the *determinant* $\det L$ of L is the absolute value of the determinant whose (i, j) element is $b_{i,j}$.

The *Euclidean norm* of a vector $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n$ is defined to be the quantity $||\mathbf{v}|| = (v_1^2 + \cdots + v_n^2)^{1/2}$ and for a polynomial $h(x, y) = \sum_{i,j} h_{i,j}x^iy^j$ the quantity $||h|| = (\sum_{i,j} |h_{i,j}|^2)^{1/2}$.

The LLL algorithm [11] acting on a matrix with rows the vectors of a basis of L and produces a basis having a quite short vector. We shall need the following result:

Lemma 1 *(LLL) Let $M = \max\{||\mathbf{b}_1||, \ldots, ||\mathbf{b}_n||\}$. The number of bit operations needed by the LLL algorithm for the computation of a vector $\mathbf{b} \in L$ with $\mathbf{b} \neq 0$ such that*

$$||\mathbf{b}|| \leq 2^{(n-1)/4}(\det L)^{1/n}$$

is $O(n^6(\log M)^3)$.

Furthermore, we shall use the following well known lemma whose proof is given in [7].

Lemma 2 *(Howgrave-Graham) Suppose $h(x, y) \in \mathbb{Z}[x, y]$ is a polynomial which is the sum of at most ω monomials. Suppose that $h(x_0, y_0) = 0 \bmod n$, where $x_0, y_0 \in \mathbb{Z}$ with $|x_0| \leq X$, $|y_0| \leq Y$ and $||h(xX, yY)|| < n/\sqrt{\omega}$. Then $h(x_0, y_0) = 0$ holds over the integers.*

Let $f(x, y) = bx + cy + dxy$ and $g(x, y) = b + cy + dxy$, where $b, c, d \in \mathbb{Z}$. Next, we give two algorithms for the solutions of the Diophantine equations $f(x, y) = 0$ and $g(x, y) = 0$, respectively.

SOLVE-CONIC1
Input: The equation $f(x,y) = 0$ and the prime factorization of b and c.
Output: The couples $(x,y) \in \mathbb{Z}^2$ with $f(x,y) = 0$.

1. Compute the sets of all the divisors $D(b)$ and $D(c)$ of b and c, respectively.
2. For every $\beta \in D(b)$ and $\gamma \in D(c)$ compute $\delta = -(b\beta + c\gamma)/d\beta\gamma$.
3. Output the couples $(\gamma\delta, \beta\delta)$ where $\beta \in D(b)$, $\gamma \in D(c)$ and δ is a positive integer.

Proof of correctness. Let $(x,y) \in \mathbb{Z}^2$ be a solution of $f(x,y) = 0$. If $\delta = \gcd(x,y)$, then $x = \delta x'$ and $y = \delta y'$, where x' and y' are integers with $\gcd(x',y') = 1$. Then, we have $bx' + cy' + d\delta x'y' = 0$. It follows that $x'|cy'$ and since $\gcd(x',y') = 1$, we get $x'|c$. Similarly, we deduce $y'|b$. Furthermore, we have $\delta = -(bx' + cy')/dx'y'$.

Time complexity. Put $M = \max\{|b|,|c|,|d|\}$. Let $|b| = p_1^{b_1} \cdots p_k^{b_k}$ be the prime factorization of b. The computation of a divisor $\beta = p_1^{\beta_1} \cdots p_k^{\beta_k}$ $(0 \leq \beta_i \leq b_i, i = 1,\dots,k)$ of b requires $O((\log\beta)^2)$ bit operations. By [6, Theorem 315], the number of positive divisors of b is $\tau(|b|) = O(|b|^\epsilon)$ for arbitrarily small $\epsilon > 0$. Hence, the computation of the set $D(b)$ requires $O(|b|^\epsilon(\log|b|)^2) = O(|b|^\epsilon)$ bit operations. Similarly, the computation of the set $D(c)$ requires $O(|c|^\epsilon)$ bit operations. Therefore, for the Step 1 we need $O(M^\epsilon)$ bit operations. For every $\beta \in D(b)$ and $\gamma \in D(c)$ the computation of δ needs $O((\log M)^2)$ bit operations and so, step 2 requires $O((\log M)^2 + M^\epsilon) = O(M^\epsilon)$ bit operations. Thus, the time complexity of SOLVE-CONIC1 is $O(M^\epsilon)$ bit operations.

Remark 1 The above algorithm implies that the number integer solutions of the equation $f(x,y) = 0$ is $O(M^\epsilon)$, where $\epsilon > 0$ is arbitrarily small.

SOLVE-CONIC2
Input: The equation $g(x,y) = 0$ and the prime factorization of b.
Output: The couples $(x,y) \in \mathbb{Z}^2$ with $g(x,y) = 0$.

1. Compute the sets of all the divisors $D(b)$ of b.
2. For every $y \in D(b)$ compute $x = -(b/y + c)/d$.
3. Output the couples (x,y) with $y \in D(b)$ and $x \in \mathbb{Z}$.

Proof of correctness. Let $(x,y) \in \mathbb{Z}^2$ be a solution of $g(x,y) = 0$. Then $y|b$ and so, $b = y\beta$, where $\beta \in \mathbb{Z}$. Simplifying the equation we get $\beta + c + dx$, whence $x = -(\beta + c)/d$.

Time complexity. Put $M = \max\{|b|,|c|,|d|\}$. The computation of $D(b)$ in Step 1 requires $O(a^\epsilon)$ bit operations, where $\epsilon > 0$ is arbitrarily small. For every $y \in D(b)$ the computation of x needs $O(M^\epsilon)$ bit operations and so Step 2 needs $O(\tau(|b|)M^\epsilon)$ bit operations. Therefore, the time complexity for the algorithm is $O(M^\epsilon)$ bit operations.

Remark 2 The above algorithm implies that the number integer solutions of the equation $g(x,y) = 0$ is $O(M^\epsilon)$, where $\epsilon > 0$ is arbitrarily small.

3 Attacks Using One Signed Message

Let $x, x' \in \{1, \ldots, q-1\}$ be such that $x = q - x'$. We set $\overline{x} = x$ if $x \leq x'$ and $\overline{x} = -x'$, otherwise. Further, we write simply $\overline{x^{-1}}$ instead of $\overline{x^{-1} \bmod q}$.

In this section we describe our attacks using one signed message. Let m be a message and (r, s) its signature with DSA (resp. ECDSA). Then there is $k \in \{1, \ldots, q-1\}$ such that $r = (g^k \bmod p) \bmod q$ (resp. $kP = (x, y)$) and $s = k^{-1}(h(m) + ar) \bmod q$.

(1) Suppose there are positive integers X and Y such that $|\overline{a}| < X$, $|\overline{k^{-1}}| < Y$ and $XY^2 < q/6^{3/2}$.

ATTACK1
Input: $(h(m), r, s)$.
Output: a.

1. Compute $b = -sr^{-1} \bmod q$ and $c = r^{-1}h(m) \bmod q$.
2. Let L be the lattice generated by $(q, 0, 0)$, $(0, qY, 0)$ and (b, cY, XY). Using the LLL algorithm, compute $(C_0, C_1, C_2) \in L$ such that $||C_0, C_1, C_2)|| \leq \sqrt{2}(q^2XY^2)^{1/3}$.
3. Compute $\gamma_1, \gamma_2 \in \mathbb{Z}$ such that $\gamma_1 = C_1/X$, $\gamma_2 = C_2/XY$ and put $\gamma_0 = C_0$.
4. Let $\Gamma(x, y) = \gamma_0 + \gamma_1 y + \gamma_2 xy$. Using the algorithm SOLVE-CONIC2, compute the set S of solutions $(x, y) \in \mathbb{Z}^2$ to $\Gamma(x, y) = 0$.
5. Compute the quantities $g^x \bmod q$ (resp. xP), where $(x, y) \in S$.
6. If $(x_0, y_0) \in S$ and $g^{x_0} = A \bmod p$ (resp. $x_0P = Q$), then output x_0. Otherwise, output "No solution".

Proof of correctness. Let $g(x, y) = b + cy + xy$. We remark that the couple $(\overline{a}, \overline{k^{-1}})$ is a solution of the congruence $g(x, y) \equiv 0 \bmod q$. Consider the polynomials $g_1(x, y) = q$ and $g_2(x, y) = qy$. The coefficient vectors of $g(xX, yY)$, $g_1(xX, yY)$ and $g_2(xX, yY)$ are $\mathbb{R}$-linearly independent and so generate a lattice L of rank 3 having as basis the rows of the matrix

$$I = \begin{pmatrix} q & 0 & 0 \\ 0 & qY & 0 \\ b & cY & XY \end{pmatrix}.$$

We have $\det L = |\det I| = q^2XY^2$. By Lemma 1, there is a vector $\mathbf{v} = (\gamma_0, \gamma_1 Y, \gamma_2 XY)$ in L such that

$$||\mathbf{v}|| \leq \sqrt{2}(q^2XY^2)^{1/3} < q/\sqrt{3}.$$

Let $\Gamma(x, y) = \gamma_0 + \gamma_1 y + \gamma_2 xy$. The polynomial $\Gamma(xX, yY)$ is an integral linear combination of $g(xX, yY)$, $g_1(xX, yY)$ and $g_2(xX, yY)$. It follows that $G(\overline{a}, \overline{k^{-1}}) \equiv 0 \pmod{q}$. Furthermore, if $\gamma_2 = 0$, then $\gamma_0 = qt_0$, $\gamma_1 = qYt_1$ with $t_0, t_1 \in \mathbb{Z}$ and so, $q\sqrt{2} \leq ||\mathbf{v}|| < q/\sqrt{3}$ which is a contradiction. Hence $c_2 \neq 0$. Since $||\Gamma(xX, yY)|| < q/\sqrt{3}$, Lemma 2 yields $\Gamma(\overline{a}, \overline{k^{-1}}) = 0$. So $(\overline{a}, \overline{k^{-1}})$

is the element of the set S of solutions $(x, y) \in \mathbb{Z}^2$ to $\Gamma(x, y) = 0$ satisfying $g^{\overline{a}} = A \bmod q$.

(2) Suppose there are positive integers X and Y such that $|\overline{k}| < X$, $|\overline{a^{-1}}| < Y$ and $XY^2 < q/6^{3/2}$. The following algorithm provide us with the secret key a.

ATTACK2
Input: $(h(m), r, s)$.
Output: a.

1. Compute $b = -rs^{-1} \bmod q$ and $c = -h(m)s^{-1}$ mod.

2. Let L be the lattice generated by $(q, 0, 0)$, $(0, qY, 0)$ and (b, cY, XY). Using the LLL algorithm, compute $(C_0, C_1, C_2) \in L$ such that $||C_0, C_1, C_2)|| \leq \sqrt{2}(q^2XY^2)^{1/3}$.

3. Compute $\delta_1, \delta_2 \in \mathbb{Z}$ such that $\delta_1 = C_1/X$, $\delta_2 = C_2/XY$ and put $\delta_0 = C_0$.

4. Let $\Delta(x, y) = \delta_0 + \delta_1 y + \delta_2 xy$. Using the algorithm SOLVE-CONIC2, compute the set S of solutions $(x, y) \in \mathbb{Z}^2$ to $\Delta(x, y) = 0$.

5. Compute the quantities $g^{y^{-1} \bmod q} \bmod p$ (resp. $(y_0^{-1} \bmod q)P$), where $(x, y) \in S$.

6. If $(x_0, y_0) \in S$ and $g^{y_0^{-1} \bmod q} = A \bmod p$ (resp. $(y_0^{-1} \bmod q)P = Q$), then output $y_0^{-1} \bmod q$. Otherwise, output "No solution".

Proof of correctness. The proof is similar to the previous one.

(3) Suppose that X and Y are positive integers such that $|\overline{k^{-1}}| < X$, $|\overline{a^{-1}}| < Y$ and $XY < q^{1/2}/6^{3/4}$. We have the following algorithm for the computation of the secret key a.

ATTACK3
Input: $(h(m), r, s)$.
Output: a.

1. Compute $b = -sh(m)^{-1} \bmod q$ and $c = rh(m)^{-1}$ mod.

2. Let L be the lattice spanned by $(qX, 0, 0)$, $(0, qY, 0)$ and (b, cY, XY). Using the LLL algorithm, compute $(C_0, C_1, C_2) \in L$ such that $||C_0, C_1, C_2)|| \leq \sqrt{2}(qXY)^{2/3}$.

3. Compute $\eta_0, \eta_1, \eta_2 \in \mathbb{Z}$ such that $\eta_0 = C_0/X$ $\eta_1 = C_1/X$ and $\eta_2 = C_2/XY$.

4. Let $H(x, y) = \eta_0 x + \eta_1 y + \eta_2 xy$. Using the algorithm SOLVE-CONIC1, compute the set S of solutions $(x, y) \in \mathbb{Z}^2$ to $H(x, y) = 0$.

5. Compute the quantities $g^{x^{-1} \bmod q} \bmod p$ (resp. $(x_0^{-1} \bmod q)P$), where $(x, y) \in S$.

6. If $(x_0, y_0) \in S$ and $g^{x_0^{-1} \bmod q} = A \bmod p$ (resp. $(x_0^{-1} \bmod q)P = Q$), then output $x_0^{-1} \bmod q$. Otherwise, output "No solution".

Proof of correctness. Let $f(x,y) = bx + cy + xy$. The couple $(\overline{a^{-1}}, \overline{k^{-1}})$ is a solution of the congruence $f(x,y) \equiv 0 \bmod q$. Let $f_1(x,y) = q$ and $f_2(x,y) = qy$. The rows of the matrix

$$J = \begin{pmatrix} qX & 0 & 0 \\ 0 & qY & 0 \\ bX & cY & XY \end{pmatrix}$$

are the coefficient vectors of $f(xX, yY)$, $f_1(xX, yY)$ and $f_2(xX, yY)$. They are $\mathbb{R}$-linearly independent and so generate a lattice L of rank 3 with $\det\Lambda = |\det J| = (qXY)^2$. By Lemma 1, there is a vector $\mathbf{v} = (\eta_0 X, \eta_1 Y, \eta_2 XY)$ in L such that

$$||\mathbf{v}|| \leq \sqrt{2}(qXY)^{2/3} < q/\sqrt{3}.$$

Put $H(x,y) = \eta_0 x + \eta_1 y + \eta_2 xy$. As in the first case we have $\eta_2 \neq 0$ and $H(\overline{a^{-1}}, \overline{k^{-1}}) = 0$. Thus, the couple $(\overline{a^{-1}}, \overline{k^{-1}})$ is an element of the set T of solutions $(x,y) \in \mathbb{Z}^2$ to $H(x,y) = 0$ satisfying $g^{\overline{a}} = A \bmod q$.

Time complexity of the attacks. We deal with the three attacks similtaneously. Step 1 needs $O((\log q)^2)$ bit operations. By Lemma 2, the application of LLL algorithm in Step 2 requires $O((\log q)^3)$ bit operations. Step 3 requires $O((\log q)^2)$ bit operations. Since the coefficients of polynomials $\Gamma(x,y)$, $\Delta(x,y)$ and $H(x,y)$ are no negarive integers $< q$, Step 4 needs $O(q^\epsilon)$ bit operations, where $\epsilon > 0$ is arbitrarily small, provided the factorization of γ_0 in the first attack, δ_0 in the second attack, and of η_0, η_1 in the third attack are known. In practice q is a prime of size 160 and so, the integers γ_0, δ_0, η_0 and η_1 have less than 50 decimal digits. As it is pointed out in [3] it is now routine to factor a 100-decimal digits integer and so, the factorization of the above numbers is quite easy with the current algorithms. Furthermore, note that the numbers γ_0, δ_0, η_0 and η_1 are random and are not constructed in a such way that their factorization is difficult. Finally, since $|S| = O(q^\epsilon)$, Step 5 needs $O(q^\epsilon(\log p)^3)$ bit operations in case of DSA and $O(q^\epsilon)$ bit operations and $O(q^\epsilon \log q)$ elliptic curve group operations in case of ECDSA. Thus, if we ignore the time needed for the factorization of γ_0, δ_0, η_0 and η_1, our attacks on DSA require $O(q^\epsilon(\log p)^3)$ bit operations and on ECDSA need $O(q^\epsilon)$ bit operations and $O(q^\epsilon \log q)$ elliptic curve group operations.

Thus we have the following theorem:

Theorem 1 *Suppose that we have a message signed with ephemeral key k and one of the following conditions holds:*
(a) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{a}| < X$, $|\overline{k^{-1}}| < Y$, $XY^2 < q/6^{3/2}$ and the factorization of γ_0 is known.
(b) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{k}| < X$, $|\overline{a^{-1}}| < Y$, $XY^2 < q/6^{3/2}$ and the factorization of δ_0 is known.
(c) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{k^{-1}}| < X$, $|\overline{a^{-1}}| < Y$, $XY < q^{1/2}/6^{3/4}$ and the factorization of η_0, η_1 is known.
Then there is a deterministic algorithm which computes a. The algorithm in case of DSA requires $O(q^\epsilon(\log p)^3)$ bit operations and in case of ECDSA needs $O(q^\epsilon)$ bit operations and $O(q^\epsilon \log q)$ elliptic curve group operations, where $\epsilon > 0$ is arbitrarily small.

4 Attacks Using Two Signed Messages

Let (r_1, s_1) and (r_2, s_2) be the DSA or ECDSA signatures of two messages m_1 and m_2 with ephemeral keys k_1 and k_2, respectively. Then we have

$$s_1 = k_1^{-1}(h(m_1) + ar_1) \bmod q \quad \text{and} \quad s_2 = k_2^{-1}(h(m_2) + ar_2) \bmod q.$$

Eliminating a from the two equalities we obtain the congruence

$$s_1 r_2 k_1 - r_1 s_2 k_2 + r_1 h(m_2) - h(m_1) r_2 \equiv 0 \pmod q.$$

Hence the couples $(\overline{k_1}, \overline{k_2^{-1}})$, $(\overline{k_1^{-1}}, \overline{k_2})$ and $(\overline{k_1^{-1}}, \overline{k_2^{-1}})$ are solutions of the congruences

$$yx + (s_1^{-1} r_1 r_2^{-1} h(m_2) - h(m_1) s_1^{-1}) y - r_1 s_2 s_1^{-1} r_2^{-1} \equiv 0 \pmod q,$$

$$yx + (s_2^{-1} r_2 r_1^{-1} h(m_1) - h(m_2) s_2^{-1}) y - r_2 s_1 s_2^{-1} r_1^{-1} \equiv 0 \pmod q,$$

$$yx + r_2 s_1 (r_1 h(m_2) - r_2 h(m_1))^{-1} y - r_1 s_2 (r_1 h(m_2) - r_2 h(m_1))^{-1} x \equiv 0 \pmod q.$$

Thus, in case we have two signed messages as above and there are positive integers X and Y satisfying one of the following:

1. $|\overline{k_1}| < X$, $|\overline{k_2^{-1}}| < Y$ and $XY^2 < q/6^{3/2}$,
2. $|\overline{k_2}| < X$, $|\overline{k_1^{-1}}| < Y$ and $XY^2 < q/6^{3/2}$,
3. $|\overline{k_2^{-1}}| < X$, $|\overline{k_1^{-1}}| < Y$ and $XY < q^{1/2}/6^{3/4}$,

we can develop similar attacks to the attacks 1, 2, 3 of Section 3, respectively. Since the algorithms of the attacks and the complexity issues are essentially the same as in the aforementioned attacks we omit their description.

We denote by $\gamma_0 + \gamma_1 y + \gamma_2 xy$, $\delta_0 + \delta_1 y + \delta_2 xy$ and $\eta_0 x + \eta_1 y + \eta_2 xy$ the polynomials constructed using the LLL-algorithm, as in the previous section, for the cases (1), (2) and (3), respectively. Then we have the following theorem:

Theorem 2 *Suppose that we have two messages signed with ephemeral keys k_1, k_2 and one of the following conditions holds:*
(a) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{k_1}| < X$, $|\overline{k_2^{-1}}| < Y$, $XY^2 < q/6^{3/2}$ and the factorization of γ_0 is known.
(b) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{k_2}| < X$, $|\overline{k_1^{-1}}| < Y$, $XY^2 < q/6^{3/2}$ and the factorization of δ_0 is known.
(c) There are integers $X > 0$ and $Y > 0$ satisfying $|\overline{k_2^{-1}}| < X$, $|\overline{k_1^{-1}}| < Y$, $XY < q^{1/2}/6^{3/4}$ and the factorization of η_0, η_1 is known.
Then there is a deterministic algorithm which computes a. The algorithm in case of DSA requires $O(q^{\epsilon}(\log p)^3)$ bit operations and in case of ECDSA needs $O(q^{\epsilon})$ bit operations and $O(q^{\epsilon}\log q)$ elliptic curve group operations, where $\epsilon > 0$ is arbitrarily small.

5 Conclusion

In this paper, combining lattice reduction techniques with algorithms for computing the integral solutions of the Diophantine equations $bx+cy+dxy=0$ and $b+cy+dxy=0$, we develop some rigorous attacks on DSA and ECDSA. If one signature is available having secret key a and ephemeral key k and each number in at least one of the sets $\{a, k^{-1} \bmod q\}$, $\{k, a^{-1} \bmod q\}$, $\{a^{-1} \bmod, k^{-1} \bmod q\}$ is quite small or quite large, then a can be computed in practice (note that if the positive integer x is large, then $\overline{x}$ is small). The same happens, if two signatures are available with ephemeral keys k_1, k_2 and each numbers in at least one of the sets $\{k_1, k_2^{-1} \bmod q\}$, $\{k_2, k_1^{-1} \bmod q\}$, $\{k_1^{-1} \bmod, k_2^{-1} \bmod q\}$ is quite small or quite large. These attacks can also be applied on other schemes where the secret and the ephemeral keys are solutions of a modular bivariate linear equation as in DSA or of a modular bivariate equation of second degree. For instance, such schemes are Schnorr' signature, Heyst-Pedersen signature, etc [12, 17].

Our attacks on DSA require $O(q^{\epsilon}(\log p)^3)$ bit operations and on ECDSA need $O(q^{\epsilon}$ bit operations and $O(q^{\epsilon}\log q)$ elliptic curve group operations. Furthermore, these two attacks need the factorization of one or two integers $< q$. Note that these numbers are random and not chosen in such a way that their factorization is difficult. In practice the size of q is 160 and so, the factorization of the above numbers is quite easy with the current algorithms. Therefore, our attacks can be quite efficient.

References

[1] M. Bellare, S. Goldwasser and Micciancio, "Pseudo-random" number generation within cryptographic algorithms: the DSS case. In *Proc. of Crypto '97,* LNCS 1294. IACR, Palo Alto, CA. Springer-Verlag, Berlin 1997.

[2] I. F. Blake and T. Garefalakis, On the security of the digital signature algorithm. *Des. Codes Cryptogr.,* 26, no. 1-3 (2002), 87-96.

[3] R. P. Brent, Recent Progress and Prospects for Integer Factorization Algorithms, D.-Z. Du et al. (Eds.): COCOON2000, LNCS 1858, pp. 3-22, Springer-Verlag Berlin Heidelberg 2000.

[4] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithm, *IEEE Transactions on Information Theory,* 31 (1985), 469-472.

[5] D. Johnson, A. J. Menezes and S. A. Vastone, The elliptic curve digital signature algorithm (ECDSA), *Intern. J. of Information Security,* 1 (2001) 36-63.

[6] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers,* Fifth edition, Oxford University Press 1979.

[7] N. A. Howgrave-Graham, *Finding small roots of univariate equations revisited.* In Cryptography and Coding, vol. 1355 of LNCS, pp. 131-142. Springer Verlag, 1997.

[8] N. A. Howgrave-Graham and N. P. Smart, Lattice Attacks on Digital Signature Schemes, *Des. Codes Cryptogr.* 23 (2001) 283-290.

[9] N. Koblitz, A. J. Menezes and S. A. Vastone, The state of elliptic curve cryptography, *Des. Codes Cryptogr.* 19 (2000), 173-193.

[10] N. Koblitz and A. J. Menezes, A survey of Public-Key Cryptosystems, *SIAM REVIEW,* 46, No. 4 (2004), 599-634.

[11] A. K. Lenstra, H. W. Lenstra Jr., and L. Lovász, Factoring polynomials with rational coefficients, *Math. Ann.*, 261 (1982), 513-534.

[12] A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography,* CRC Press, Boca Raton, Florida, 1997.

[13] National Institute of Standards and Technology (NIST). *FIPS Publication* 186*: Digital Signature Standard.* May 1994.

[14] P. Nguyen and I. E. Shparlinski, The Insecurity of the Digital Signature Algorithm with Partially Known Nonces, *J. Cryptology,* 15 (2002), 151-176.

[15] 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.

[16] D. Poulakis, A variant of Digital Signature Algorithm, *Des. Codes Cryptogr.* 51, No. 1 (2009), 99-104.

[17] D. R. Stinson, *Cryptography, Theory and Practice,* Chapman & Hall/CRC, 2nd ed. 2002.