International Journal of Pure and Applied Mathematics
Volume 70 No. 7 2011, 939-949

CONDITIONS ON THE GENERATOR FOR FORGING ELGAMAL SIGNATURE

Omar Khadir
Laboratory of Mathematics, Cryptography and Mechanics
University of Hassan II
Mohammedia, Casablanca, Fstm, MOROCCO

Abstract: This paper describes new conditions on parameters selection that lead to an efficient algorithm for forging ElGamal digital signature. Our work is inspired by Bleichenbacher's ideas.

AMS Subject Classification: 94A60
Key Words: public key cryptography, discrete logarithm problem, ElGamal digital signature

1. Introduction

Numerous digital signature algorithms have been developed since the invention of the public key cryptography in the late 1970s, see [3], [15], [14]. They almost all have the same principle. Every user possesses two kinds of keys. The first one is private, must be kept secret and stored only locally. The second is public and must be largely diffused to be accessible to the others users. To sign a particular message, a contract or a will M, Alice has to solve a hard mathematical equation depending of M and of her public key. With the help of her private key, she is able to furnish the solutions. Bob, the judge or anybody, can verify that the solutions computed by Alice are valid. For an adversary, without knowing Alice private key, the algorithm is constructed in such a way that it is computationally too hard to solve the considered equation.

One of the most popular signature algorithm was proposed by ElGamal [4]. It has many variants (see [16], [17], [9], [5]) and is based on the hard discrete logarithm problem. Since its conception in 1985, several attacks were mounted

Received: March 14, 2011

and have revealed possible weaknesses if the signature keys were not carefully selected, see [2], [1], [10], [13]. However, no general method for breaking totally the system is known.

In an ElGamal signature protocol, a signer, in addition to his private key, must detain three other integer parameters (p, α, y) as a public key. In 1996, Bleichenbacher [2] presented a cryptanalysis where he showed that if the generator α and the modulus p verify some special relations, it is possible to forge ElGamal signature for any arbitrary message. In particular, he proved that, the signature scheme becomes insecure when parameters α and p are chosen such that α divides $p-1$. Hence, selecting $\alpha = 2$ is imprudent.

The purpose of our work, is to describe new conditions on parameters selection that lead to an efficient algorithm for forging ElGamal signature for any arbitrary message. As an extension of Bleichenbacher's result, we show that, if the modular inverse of the generator α divides $p-1$, then it is possible to break the system. As an example, the choice of $\alpha = \frac{p+1}{2}$ as a generator, is not recommended.

The paper is organized as follows. Section 2 contains preliminaries which will be utilized in the sequel. Our contribution, mainly composed by Algorithm 2 and Corollary 3, is presented in Section 3. We conclude in Section 4.

Throughout this article, we will adopt ElGamal paper notations, see [4]. $\mathbb{Z}$, $\mathbb{N}$ are respectively the sets of integers and non-negative integers. For every positive integer n, we denote by $\mathbb{Z}_n$ the finite ring of modular integers and by $\mathbb{Z}_n^*$ the multiplicative group of its invertible elements. Let a, b, c be three integers. The great common divisor of a and b is denoted by $gcd(a, b)$. We write $a \equiv b$ $[c]$ if c divides the difference $a - b$, and $a = b \ mod \ c$ if a is the remainder in the division of b by c. The positive integer a is said to be B-smooth (see [8], p. 92), $B \in \mathbb{N}$, if every prime factor of a is less than or equal to the bound B.

We start, in the next section, by preliminaries containing known mathematical facts that will be exploited later.

2. Preliminaries

Before exploring new situations under which one can forge ElGamal digital signature, we briefly review three questions that are directly related to our result.

2.1. Discrete Logarithm Problem when $p-1$ is B-Smooth

The discrete logarithm problem importance started to grow with the publication in 1976 of the fundamental work of Diffie and Hellman [3], [11]. The issue became central in public key cryptography.

Let p be a prime integer and α a primitive root of $\mathbb{Z}_n^*$. We consider the discrete logarithm equation

$$\alpha^x \equiv y\ [p] \tag{1}$$

where y is fixed in $\{1, 2, 3, \ldots, p-1\}$, and x is unknown in $\{0, 1, 2, \ldots, p-2\}$.

In 1978, Pohlig and Hellman [12] published a practical method to solve equation (1) when all the prime factors of $p-1$ are not too large. Let us recall the outlines of their algorithm.

Assume that $p-1$ is B-smooth. The bound B depends on the computers power. This implies that we can obtain the prime factorization of $p-1$: $p-1 = p_1^{n_1} p_2^{n_2} \ldots p_k^{n_k}$ where $n_i, k_i \in \mathbb{N}^*$ for $1 \leq i \leq k$. We will first find x modulo $p_i^{n_i}$ for every $i \in \{1, 2, \ldots, k\}$ and apply the Chinese Remainder Theorem (see [8], p. 68) to compute x modulo $p_1^{n_1} p_2^{n_2} \ldots p_k^{n_k}$.

The p_1-ary representation of $x_1 = x\ mod\ p_1^{n_1}$ can be written as:

$$x_1 = b_0 + b_1 p_1 + \ldots + b_{n_1-1} p_1^{n_1-1} \tag{2}$$

where $b_0, b_1, \ldots, b_{n_1-1}$ are unknown in $\{0, 1, \ldots p_1 - 1\}$.

Let $\lambda_1 = p_1^{n_1-1} p_2^{n_2} \ldots p_k^{n_k}$. We have $\lambda_1 x_1 = \lambda_1 b_0 + K_1 (p-1)$, $K_1 \in \mathbb{N}$. Since $\alpha^{K_1 (p-1)} \equiv 1\ [p]$, equation (1) can be transformed to $\alpha^{\lambda_1 x_1} \equiv y^{\lambda_1}\ [p]$ and therefore

$$\alpha^{\lambda_1 b_0} \equiv y^{\lambda_1}\ [p] \tag{3}$$

From equation (3) we obtain the first coefficient b_0.

Similarly, if $\lambda_2 = p_1^{n_1-2} p_2^{n_2} \ldots p_k^{n_k}$, then $\lambda_2 x_1 = \lambda_2 b_0 + \lambda_2 b_1 p_1 + K_2 (p-1)$, $K_2 \in \mathbb{N}$. As $\alpha^{K_2 (p-1)} \equiv 1\ [p]$, equation (1) becomes $\alpha^{\lambda_2 x_1} \equiv y^{\lambda_2}\ [p]$ and therefore

$$\alpha^{\lambda_2 b_0 + \lambda_2 b_1 p_1} \equiv y^{\lambda_2}\ [p] \tag{4}$$

We get the second coefficient b_1 from equation (4).

Gradually, we compute $b_0, b_1, b_2, \ldots, b_{n_1-1}$ and then determine x_1. This is computationally possible since $2^{n_1} \leq p-1$ and then $n_1 \leq \frac{\ln(p-1)}{\ln 2}$. In other word, n_1 is bounded by the bit length of $p-1$.

We repeat the technique with $p_2, p_3, \ldots, p_k$ and arrive to the following system of congruences:

$$\begin{cases} x \equiv x_1 & [p_1^{n_1}] \\ x \equiv x_2 & [p_2^{n_2}] \\ \quad \vdots & \quad \vdots \\ x \equiv x_k & [p_k^{n_k}] \end{cases} \tag{5}$$

Natural index k is not too large since $k \leq \sum_{i=1}^{k} n_i \leq \frac{\ln(p-1)}{\ln 2}$. Hence, system (5) can efficiently be solved by the Chinese Remainder Theorem method whose running time is $O(\ln^2 p)$ bit operations. The complexity of Pohlig-Hellman algorithm is $O(\sum_{i=1}^{k} n_i(\ln p + \sqrt{p_i}))$ bit operations, see [12], [9], p. 108.

2.2. ElGamal Signature Algorithm

We recall the basic ElGamal protocol (see [4], [19], [8]) in three steps.

1. Alice begins by choosing three numbers p, α and x such that:

— p is a large prime integer.

— α is a primitive root of the finite multiplicative group $\mathbb{Z}_p^*$.

— x is a random element taken in $\{1, 2, \ldots, p-2\}$.

She computes $y = \alpha^x \; mod \; p$ and publishes the triplet (p, α, y) as her public key. She keeps secret the parameter x as her private key.

2. Suppose that Alice desires to sign the message $m < p$. She must solve the modular equation

$$\alpha^m \equiv y^r \, r^s \; [p] \tag{6}$$

where r and s are two unknown variables.

Alice computes $r = \alpha^k \; mod \, p$, where k is selected randomly and is invertible modulo $p-1$. She has exactly $\varphi(p-1)$ possibilities for k, where φ is the phi-Euler function. Equation (6) is then equivalent to:

$$m \equiv x\, r + k\, s \; [p-1] \tag{7}$$

As Alice possesses the secret key x, and as the integer k is invertible modulo $p-1$, she computes the second unknown variable s from relation (7) by:

$$s \equiv \frac{m - x\, r}{k} \; [p-1] \tag{8}$$

The inverse modulo $p-1$ of the integer k in equation (8) is computed by the extended Euclidean algorithm whose complexity is $O(\ln^2 p)$ bit operations.

3. Bob can verify the signature by checking that congruence (6) is valid.

Observe that, in step **1.**, we need to know how to construct signature keys. Generally, the running time for generating prime integers takes the most important part in the total running time. In [6], we made experimental tests and concluded by suggesting some rapid procedures. In step **2.**, the random integer k must be kept secret, otherwise relation (7) allows any adversary to obtain Alice private key x. The fact of having many possibilities for the valid pairs (r, s) does not affect the system security. Indeed, these pairs are uniformly distributed, see [18].

To prevent obvious attacks against ElGamal signature scheme, some of theme mentioned in the original paper [4], it is necessary to work with a free collision hash function h. The message M is simply replaced by $m = h(M)$ before applying the signature algorithm. We can take h equal to the secure hash algorithm SHA1, see [19], p. 139, Chapter 9.

2.3. Bleichenbacher's Attack

In Eurocrypt'96 meeting, Bleichenbacher indicated a possible weakness in the ElGamal signature scheme if the keys are not properly chosen [2], [1]. His ideas are summarized in his main theorem.

Theorem 1. (see [2]) *Let $p-1 = b\,w$ where b is B-smooth and let $y_A \equiv \alpha^{x_A}$ $[p]$ be the public key of user A. If a generator $\beta = c\,w$ with $0 < c < b$ and an integer t are known such that $\beta^t \equiv \alpha$ $[p]$ then a valid signature (r, s) on a given h can be found.*

This theorem has the immediate consequence:

Corollary 1. (see [2]) *If α is B-smooth and divides $p-1$, then it is possible to generate a valid ElGamal signature on an arbitrary value h.*

Now, we can move to the next section where we expose our contribution.

3. Our Contribution

In this section, we present our main result. The first sufficient condition for forging ElGamal signature is based on a slight simplification of Bleichenbacher's theorem. More precisely, we have:

Theorem 2. *Let (p, α, y) be Alice public key in an ElGamal signature scheme. If an adversary can compute a nonnegative integer $k \le p-2$, relatively* *prime to $p-1$ and such that $\frac{p-1}{gcd(p-1,\alpha^k \ mod \ p)}$ is B-smooth, then he will be able to forge Alice signature.*

Proof. We follow the method used in [2]. Let (p,α,y) be Alice public key in an ElGamal signature protocol. If we put $D = gcd(p-1,\alpha^k \ mod \ p)$, then there exist two coprimes p_1 and a_1 such that $p-1 = D\,p_1$ and $\alpha^k \ mod \ p = D\,a_1$. Let H be the subgroup of $\mathbb{Z}_p^*$ generated by the particular element $\alpha^D \, mod \ p$. Since $y \equiv \alpha^x \, [p]$, where the natural integer x is Alice secret key, we have $y^D \equiv (\alpha^D)^x \, [p]$ and then $y^D \, mod\, p \in H$. By hypothesis, the order of the subgroup H, $\frac{p-1}{D} = p_1$ is B-smooth, so the discrete logarithm equation

$$(\alpha^D)^X \equiv y^D \, [p] \tag{9}$$

can be solved in polynomial time. Hence there exists $x_0 \in \mathbb{N}$ such that $(\alpha^D)^{x_0} \equiv y^D \, [p]$. To forge Alice signature for a message M, and a hash function h, if the adversary puts $m = h(M)$, he must find two positive integers r, s such that $\alpha^m \equiv y^r \, r^s \, [p]$. If he chooses $r = \alpha^k \, mod\, p$, he will have the equivalences: $\alpha^m \equiv y^r \, r^s \, [p] \Longleftrightarrow \alpha^m \equiv y^{D\,a_1} \, \alpha^{k\,s} \, [p] \Longleftrightarrow \alpha^m \equiv [(\alpha^D)^{x_0}]^{a_1} \, \alpha^{k\,s} \, [p] \Longleftrightarrow \alpha^m \equiv \alpha^{x_0\,r} \, \alpha^{k\,s} \, [p]$
$\Longleftrightarrow m \equiv x_0\,r + k\,s \, [p-1] \Longleftrightarrow s \equiv \frac{m - x_0\,r}{k} \, [p-1]$.

So (r,s) is a valid signature for the message M, obtained without knowing Alice secret key. □

If the first valid exponent k in our Theorem 2 is not too large, the adversary can construct the following deterministic algorithm in order to forge ElGamal signature. Consequently, we recommend that when selecting the signature keys, we have to verify that for any $k \leq K_0$, where $gcd(k, p-1) = 1$ and K_0 is the largest bound allowed by computer power, the integer $\frac{p-1}{gcd(p-1,\alpha^k \ mod \ p)}$ has at least one large prime factor.

Algorithm 1.
Input: Alice public key (p,α,y) and the message M to be signed.
Output: The signature (r,s) of M.

1. Read(p,α,y); $\{(p,\alpha,y)$ is Alice public key$\}$.
2. Read(M); m:=h(M); $\{m$ is the hashed of the message to be signed$\}$.
3. $j \leftarrow -1$; {Initialization of integers j which play the role of exponents k}.
4. $F \leftarrow 0$; $\{F$ is a flag$\}$.
5. While (F=0) do
  5.1. $j \leftarrow j+2$; $\{j$ must be invertible modulo $p-1$, so j is odd$\}$.
  5.2. If $gcd(p-1, j) = 1$ then
    5.2.1. $r \leftarrow \alpha^j \ mod \ p$; $\{$ r will be the first parameter of the signature$\}$.

5.2.2. $D \leftarrow gcd(p-1, r)$; { D will be the exponent in equation (9)}.
5.2.3. If $(p-1)/D$ is B-smooth, then
5.2.3.1 $k \leftarrow j$; { k is the searched exponent of α}.
5.2.3.2 $F \leftarrow 1$; { To stop the while loop}.
6. $x_0 \leftarrow X$; {X is a solution of equation (9), obtained by Pohlig-Hellman algorithm, see [12]}.
7. $s \leftarrow \frac{m - x_0\, r}{k}\ mod\ (p-1)$; {$s$ is the second parameter of the signature}.
8. Return(r, s); {(r, s) is the digital signature}.

Our theorem 2 has a first remarkable consequence: if Q denotes the part of the prime factorization of $p-1$ that is not B-smooth and if $\alpha^k\ mod\ p$, for some $k \in \mathbb{N}$, is a multiple of Q then ElGamal signature scheme is insecure. More formally:

Corollary 2. *Let $p_1^{n_1}\, p_2^{n_2} \ldots p_k^{n_k}\, q_1^{n'_1}\, q_2^{n'_2} \ldots q_l^{n'_l}$ be the classical prime factorization of $p-1$, where $p_1^{\alpha_1}\, p_2^{\alpha_2} \ldots p_r^{\alpha_r}$ is B-smooth. If an adversary can compute a natural integer k, $k \leq p-2$, relatively prime to $p-1$ and such that $\alpha^k\ mod\ p$ is a multiple of $Q = q_1^{n'_1}\, q_2^{n'_2} \ldots q_l^{n'_l}$, then he will be able to forge Alice signature for any arbitrary message M.*

Proof. Observe, first, that α is not necessary a divisor of $p-1$, nor a B-smooth integer. Since $\alpha^k\ mod\ p$ is a multiple of Q, we have Q divides $gcd(p-1, \alpha^k\ mod\ p)$. So $p_1^{n_1}\, p_2^{n_2} \ldots p_k^{n_k}$ is a multiple of $\frac{p-1}{gcd(p-1, \alpha^k\ mod\ p)}$, and then $\frac{p-1}{gcd(p-1, \alpha^k\ mod\ p)}$ is B-smooth. We conclude by applying Theorem 2. $\square$

If the first valid exponent k is not too large, Corollary 2 leads to a more practical algorithm for forging ElGamal signature.

Algorithm 2.
Input: Alice public key (p, α, y) and the message M to be signed.
Output: The signature (r, s) of M.

1. Read(p, α, y); {(p, α, y) is Alice public key}.
2. Read(M); m:=h(M); {m is the hashed of the message to be signed}.
3. $Q_0 \leftarrow Q$; {Q is the part of $p-1$ that is not B-smoot}.
4. $j \leftarrow -1$; {Initialization of integers j which play the role of exponents k}.
5. $F \leftarrow 0$; {F is a flag}.
6. While (F=0) do
6.1. $j \leftarrow j+2$; {j must be invertible modulo $p-1$, so j is odd}.
6.2. If $gcd(p-1, j) = 1$; then
6.2.1. $r \leftarrow \alpha^j\ mod\ p$; { r will be the first parameter of the signature}.
6.2.2. If $r\ mod\ Q_0 = 0$ then

6.2.2.1. $k \leftarrow j$; { k is the searched exponent of α}.
6.2.2.2. $F \leftarrow 1$; { To stop the while loop}.
7. $D \leftarrow gcd(p-1, r)$; { D will be the exponent in equation (9)}.
8. $x_0 \leftarrow X$; {X is a solution of equation (9), obtained by Pohlig-Hellman algorithm, see [12]}.
9. $s \leftarrow \frac{m - x_0 r}{k}$ mod $(p-1)$; {s is the second parameter of the signature}.
10. Return(r, s); {(r, s) is the digital signature}.

Next result, which can be seen as an extension of Blechenbacher's Corollary 1, shows that in an ElGamal signature scheme, it is not secure to have a primitive root whose modular inverse divides $p-1$. In particular, as a primitive root, $\alpha = \frac{p+1}{2}$ is not recommended since its inverse is 2. More explicitly:

Corollary 3. *Let (p, α, y) be Alice public key in an ElGamal signature protocol. An adversary can forge Alice signature for any given message if one of the following conditions is satisfied:*

a) $p \equiv 1$ [4], α is B-smooth and divides $p-1$.

b) $p \equiv 1$ [4], $\frac{1}{\alpha}$ mod p is B-smooth and divides $p-1$.

c) α^2 is B-smooth and divides $p-1$.

Proof. a) Put $p - 1 = \alpha Q$. As α is a primitive root, we have $\alpha^{(p-1)/2} \equiv -1$ $[p]$, and so $\alpha^k \equiv Q$ $[p]$ where $k = (p-3)/2$. Consequently $gcd(p-1, \alpha^k \ mod \ p) = Q$ and then $\frac{p-1}{gcd(p-1, \alpha^k \ mod \ p)} = \alpha$ which is smooth and this allows the use of our theorem 2.

b) It is easy to see that α is a primitive root modulo p if and only if $\frac{1}{\alpha}$ mod p is a primitive root. Suppose that $\frac{1}{\alpha}$ mod p is B-smooth and divides $p-1$. If the public key was $(p, \frac{1}{\alpha} \ mod \ p, \frac{1}{y} \ mod \ p)$, and the private key was the same parameter x, an adversary would be able to forge the signature and to find two valid integers (r_1, s_1) for any arbitrary message M. With $(r, s) = (r_1, -s_1 \ mod \ p-1)$, the adversary forges Alice signature for the message M. Indeed, ElGamal equation (1) is equivalent to

$$(\frac{1}{\alpha})^m \equiv (\frac{1}{y})^{r_1} (r_1)^{-s_1} \ [p]. \tag{10}$$

c) If $p \equiv 1$ [4], then the affirmation is true from case a). Assume then that $p \equiv 3$ [4] and put $p = 3 + 4K$, $K \in \mathbb{N}^*$. As α is a primitive root, we have $\alpha^{(p-1)/2} \equiv -1$ $[p]$, and then $\alpha^2 (\alpha^{(p-5)/2} \ mod \ p) = p - 1$. Let $k = (p-5)/2$. Since $k = 2K - 1$, $gcd(p-1, k)$ divides 4 and as k is odd, $gcd(p-1, k) = 1$. On the other hand $\frac{p-1}{gcd(p-1,\alpha^k \ mod \ p)} = \frac{p-1}{\alpha^k \ mod \ p} = \alpha^2$ is B-smooth which allows us to apply theorem 2 and achieve the proof. $\square$

Before concluding, we give the following theoretical theorem relative to the number of exponents k figuring in Corollary 2 and Algorithm 2.

Theorem 3. *Let α be a primitive root of the multiplicative group $\mathbb{Z}_p^*$. For any fixed integer Q such that $1 \leq Q \leq p-1$, if we set*

$$E_\alpha = \{k \in \mathbb{N} \ / \ 1 \leq k \leq p-2, \ gcd(p-1,k) = 1, \text{ and } Q \text{ divides } \alpha^k \ mod \ p\} \quad (11)$$

then the cardinality of E_α is independent of the choice of the primitive root α.

Proof. Let α, β be two fixed primitive roots of $\mathbb{Z}_p^*$. It is well-known that there exists $i \in \{1, 2, \ldots, p-2\}$ such that $gcd(p-1, i) = 1$ and $\alpha \equiv \beta^i \ [p]$. Consider then the function:

$$f \ : \ E_\alpha \longrightarrow E_\beta,$$
$$k \mapsto ik \ mod \ (p-1).$$

First we have $f(k) \in E_\beta$, where E_β is defined like E_α in relation (13). Indeed $gcd(p-1, ik \ mod \ (p-1)) = gcd(p-1, ik) = 1$ since $gcd(p-1,k) = gcd(p-1,i) = 1$.

On the other hand $\beta^{ik \ mod \ (p-1)} \equiv \beta^{ik} \equiv \alpha^k \ [p]$, so Q divides $\beta^{ik \ mod \ (p-1)}$ $mod \ p$ and therefore $ik \ mod \ (p-1) \in E_\beta$.

Let us now establish that f is an injective function. We have successively: $f(k) = f(k') \Longrightarrow ik \ mod \ (p-1) = ik' \ mod \ (p-1) \Longrightarrow i\,k \equiv ik' \ [p-1] \Longrightarrow k \equiv k' \ [p-1]$ since i is invertible modulo $p-1$. As $1 \leq k, k' \leq p-2$, we obtain that $k = k'$.

Since f is injective $Card(E_\alpha) \leq Card(E_\beta)$ and by interchanging the role of the parameters α and β, we find that $Card(E_\beta) \leq Card(E_\alpha)$ and so $Card(E_\alpha) = Card(E_\beta)$. This means that $Card(E_\alpha)$ is a constant depending only of the two integers p and Q. $\square$

4. Conclusion

In this paper, we described new conditions on parameters selection that can lead to an efficient deterministic algorithm for forging ElGamal digital signature. Our approach is based on the work of Bleichenbacher presented at Eurocrypt'96 conference, see [2].

References

[1] R. Anderson, S. Vaudenay, Minding Your P's and Q's, In: *Advances in Cryptology, Asiacrypt'96, LNCS 1163*, Springer-Verlag (1996), 26-35.

[2] D. Bleichenbacher, Generating ElGamal signatures without knowing the secret key, In: *Advances in Cryptology, Eurocrypt'96, LNCS 1070*, Springer-Verlag (1996), 10-18.

[3] W. Diffie, M.E. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, **IT-22** (1976), 644-654.

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

[5] P. Horster, M. Michels, H. Petersen, *Generalized ElGamal Signature Schemes for One Message Block*, Technical Report, TR-94-3 (1994).

[6] O. Khadir, L. Szalay, Experimental results on probable primality, *Acta Univ. Sapientiae, Math.*, **1**, No. 2 (2009), 161-168; http://www.emis.de/journals/AUSM/C1-2/math2-6.pdf

[7] O. Khadir, New variant of ElGamal signature scheme, *Int. J. Contemp. Math. Sciences*, **5**, No. 34 (2010), 1653-1662; http://www.m-hikari.com/forth2/khadirIJCMS33-36-2010.pdf

[8] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, Florida (1997); http://www.cacr.math.uwaterloo.ca/hac/

[9] National Institute of Standard and Technology (NIST). *FIPS Publication 186*, Digital Signature Algorithm, Department of Commerce (1994); http://www.itl.nist.gov/fipspubs/fip186.htm

[10] P.Q. Nguyen, I.E. Shparlinski, The insecurity of the digital signature algorithm with partial known nonces, *J. of Cryptology*, **15** (2002), 151-176.

[11] A. Odlyzko, M.E. Hellman, Discrete logarithms, the past and the future, *J. of Designs, Codes and Cryptography*, **19** (2000), 129-145.

[12] S.C. Pohlig, M.E. Hellman, An improved algorithm for computing logarithms over GF(p) and its cryptographic significance, *IEEE Trans. Information Theory*, **IT-24**, No. 1 (1978), 106-110.

[13] D. Pointcheval, J. Stern, Security argument for digital signatures and blind signatures, *J. of Cryptology*, **13** (2000), 361-396.

[14] M.O. Rabin, Digitalized signatures and public key functions as intractable as factoring, *MIT/LCS/TR*, **212** (1979).

[15] R. Rivest, A. Shamir, L. Adeleman, A method for obtaining digital signatures and public key cryptosystems, *Communication of the ACM*, **21** (1978), 120-126.

[16] C.P. Schnorr, Efficient identification and signatures for smart cards, In: *Advances in Cryptology, Crypto'89, LNCS 435*, Springer-Verlag (1990), 239-252.

[17] C.P. Schnorr, Efficient signature generation by smart cards, *J. of Cryptology*, **4** (1991), 161-174.

[18] I.E. Shparlinski, On the uniform distribution of the ElGamal signature, *AAECC*, Springer-Verlag, **13** (2002), 9-16.

[19] D.R. Stinson, *Cryptography, Theory and Practice*, Third Edition, Chapman and Hall/CRC (2006).