Quasigroups and Related Systems **20** (2012), 261 – 266

New signature scheme based on difficulty of finding roots

Nikolai A. Moldovyan and Victor A. Shcherbacov

Abstract. There are considered two digital signature schemes based on difficulty of finding the wth roots in the finite ground fields $GF(p)$. The first scheme uses the prime value $p = Nt_0t_1t_2+1$, where N is an even number; t_0, t_1, t_2 are prime numbers such that $|t_0| \approx |t_1| \approx |t_2| \approx 80$ bits. The public key is defined as follows $Y = K_1^{w_1}K_2^{w_2}$, where $w_1 = t_0t_1$ and $w_2 = t_0t_2$. The second scheme uses the value $p = Nt_1t_2 + 1$, and the public key composed of two values $Y_1 = K_1^{t_1}K_2^{t_2} \mod p$ and $Y_2 = K_3^{t_1}K_4^{t_2} \mod p$, where four numbers K_1, K_2, K_3, and K_4 are the private key.

1. Introduction

There are well known signature schemes based on the difficulty of finding discrete logarithms [1] and factorization [3, 6] problems.

In paper [2] it has been proposed the signature scheme based on difficulty of finding the kth roots in the finite fields $GF(p)$ such that $p = Nk^2 + 1$, where k is sufficiently large prime having the size $| k | \geqslant 160$ bits and N is even number such that the size of p is $| p | \geqslant 1024$ bits.

To provide faster signature generation and verification procedures it is interesting to design signature schemes based on the last problem defined over the elliptic curves (ECs) [5] having the order divisible by the square of large prime k. However generating the EC with required order is an open problem. In the present paper there are considered other approaches to designing signature schemes based on difficulty of finding roots in the finite ground fields. The proposed approaches can be applied with using the ECs.

2. The first signature scheme

2.1. Algorithms for signature generation and verification

For the synthesis of the DS schemes it can be used complexity of finding the roots of large degree modulo prime p in the case of the modulus structure $p = Nt_0t_1t_2 + 1$,

2010 Mathematics Subject Classification: 11G20 11T71
Keywords: cryptography, public key, digital signature, difficult problem, computing roots

where N is even number; t_0, t_1, t_2 are prime numbers such that $|t_0| \approx |t_1| \approx |t_2| \approx$ 80 bits. In such signature schemes the difficulty of finding the wth roots is defined by difficulty of performing large number of checks that are required to find a value that can be represented as the wth power of some number.

It is supposed performing computations in the multiplicative group of the finite ring $(Z_p, +, \cdot)$. The security of the DS scheme using the prime modulus $p = Nt_0t_1t_2 + 1$ is defined by the fact that procedure of finding the qth roots, where q is a prime that divides the group order Ω, can be performed only for Ω/q different elements of the group. For sufficiently large value q probability that a random element a can be represented as x^q is negligible. Let us consider the construction of the DS scheme.

The public key Y is formed using two private keys $K_1 < p$ and $K_2 < p$ that are selected at random. The public key is calculated as follows $Y = K_1^{w_1} K_2^{w_2}$, where $w_1 = t_0t_1$ and $w_2 = t_0t_2$. This is a characteristic feature of the considered signature scheme. The digital signature is a triple e, S_1 and S_2. Suppose a message M is given. The signature generation procedure is performed as follows:

1. Select at random two numbers T_1 and T_2.
2. Calculate the value $R = T_1^{w_1} T_2^{w_2} \pmod p$.
3. Calculate the first signature element e: $e = F(R, M) = RH \pmod{w_1}$, where H is the hash value computed from the message: $H = F_H(M)$.
4. Calculate the second signature element S_1 using the formula $S_1 = T_1 K_1^{-e} \pmod p$.
5. Calculate the third signature element S_2 using the formula $S_2 = T_2 K_2^{-e} \pmod p$.

2.2. The signature verification algorithm

The signature verification algorithm is as follows.

1. Using the given signature (e, S_1, S_2) calculate the value $R' = Y^e S_1^{w_1} S_2^{w_2} \pmod p$.
2. Calculate the value $e' = F(R', M) = R'H \pmod{w_1}$.
3. Compare e' with e. If $e' = e$, then the signature is valid.

Proof that signature verification works. If the digital signature has been formed correctly, i.e., using the true private key in accordance with the specified procedure for the signature generation, then in step 3 of the signature verification procedure it is obtained the equality of the values e and e'. On the basis of the equality $e = e'$ it is concluded that the signature is valid. Correctness of the signature scheme can be shown as follows. Substituting into the formula $R' = Y^e S_1^{w_1} S_2^{w_2} \pmod p$ the values $Y = K_1^{w_1} K_2^{w_2} \pmod p$, $S_1 = T_1 K_1^{-e} \pmod p$, and $S_2 = T_2 K_2^{-e} \pmod p$ we obtain:

$$R' = (K_1^{w_1} K_2^{w_2})^e \, (T_1 K_1^{-e})^{w_1} \, (T_2 K_2^{-e})^{w_2} \pmod p = T_1^{w_1} T_2^{w_2} \pmod p = R,$$

i.e., the value R' obtained at the first step of the signature verification procedure is equal to R, therefore $e' = R'H \pmod{w_1} = RH \pmod{w_1} = e$.

2.3. Possible attacks

Let us consider some attacks on the constructed signature algorithm.

The first type attack. In the first attack it is supposed that a potential attacker can do the following attack, including the generation of random values T_1 and T_2, then calculate value $R = T_1^{w_1} T_2^{w_2} \pmod p$, $e = F(R, M)$ and try to find a pair numbers S_1 and S_1 such that the following equation $S_1^{w_1} S_2^{w_2} = RY^{-e} \pmod p$ holds, where S_1 and S_2 are the unknowns.

In this case the right side of the equation has a random value because a function $F(R, M)$ is a confusion function, for example, a hash function, or function of the form $e = RH \pmod{w_1}$.

If you set one of the unknowns, for example S_2, the equation is transformed into an equation with unknown S_1. In the last equation the right side with negligibly small probability will have a value, at which the last equation is solvable. An exponentiation operation modulo p is performed to verify condition of the solvability. To obtain the case when the solvability condition is satisfied, it is required to perform the described attempt on the average t_1 time. When the length of t_1 equal to 80 bits or more, the computational complexity of forging the signature is so high that it is practically infeasible. Similarly, the signature forgery can be performed with solving some equation relatively unknown S_2, when it is required do t_2 described attempts. If the length of the value t_2 is equal to 80 bits or more, then the computational difficulty of such attempt is sufficiently high and the attack is infeasible.

The second type attack. The second attack model is more sophisticated. In the second variant it is considered the case in which the attacker generates the value $R = Y^u \pmod p$, calculates $e = F(R, M)$, and tries to find a pair of the numbers S_1 and S_2 using the formulas $S_1 = Y^{s_1 w_1} \pmod p$ and $S_2 = Y^{s_2 w_2} \pmod p$.

For this representation of the desired values S_1 and S_2 the expression $Y^u = Y^e Y^{s_1 w_1} Y^{s_2 w_2} \pmod p$ holds, if the following relation holds $u - e = s_1 w_1 + s_2 w_2 \pmod{(p-1)}$, which is a Diophantine equation for the unknown s_1 and s_2.

Because $w_1 = t_0 t_1$ and $w_2 = t_0 t_2$, where t_0, t_1, and t_2 are prime numbers, then this Diophantine equation has a solution in integers only in the case when the right side of the equation is divisible by the number t_0, which is equal to the greatest common divisor of the coefficients for the unknowns s_1 and s_2. The value e is determined by the formula $e = F(R, M)$ and has a random value. The probability that a number t_0 will divide the number $u - e$ (i.e., the probability that a Diophantine equation has solutions) is $1/t_0$.

When the size t_0 is equal to 80 bits, for one case the solvability of the Diophantine equation requires on the average to perform 2^{80} attempts to forge the signature. The difficulty of the last process exceeds 2^{80} exponentiations modulo p.

The third type attack. The most effective method for attacking the signature scheme is based on solving the discrete logarithm problem in the finite field $GF(p)$.

The method is described as follows. It is easy to find a primitive element G, the degree of which run through all nonzero elements of the field $GF(p)$. Then the public key can be represented as:

$$Y = G^z = X_1^{w_1} X_2^{w_2} = G^{x_1 w_1} G^{x_2 w_2} = G^{x_1 w_1 + x_2 w_2} \pmod{p},$$

where x_1 and x_2 are the values of the discrete logarithms of the secret key elements X_1 and X_2, respectively. The last relation shows that finding the discrete logarithm z from the public key to the base G allows one to obtain the equation $z = x_1 w_1 + x_2 w_2 = x_1 t_0 t_1 + x_2 t_0 t_2 \pmod{(p-1)}$.

The last equation can be easily solved relatively the unknowns x_1 and x_2. Its solvability follows from the fact of the divisibility of numbers z by t_0. Let $z = z' t_0$. Then we have $z' = x_1 t_1 + x_2 t_2 \pmod{(p-1)/t_0}$

From the last relation, for some integer N we obtain the following equation with two unknowns x_1 and x_2: $z' + N\frac{p-1}{t_0} = x_1 t_1 + x_2 t_2$, from which it follows

$$z' = x_1 t_1 \pmod{t_2} \Rightarrow x_1 = \frac{z'}{t_1} \pmod{t_2}$$

Similarly, one can obtain a formula for calculating the second unknown x_2: $x_2 = \frac{z'}{t_2} \pmod{t_1}$. Thus, the DS scheme proposed in this section requires to use a prime p, whose size is not less than 1024 bits. In the last case the discrete logarithm problem can be considered as practically infeasible one, since its difficulty estimation is 2^{80} multiplications $\mod p$ [4]. Thus, the proposed signature scheme provides security $\geqslant 2^{80}$ for values p having size $\geqslant 1024$ bits.

3. The second signature scheme

3.1. Algorithms for generation and verification signatures

Let us consider another variant of the construction of the DS scheme based on difficulty of finding the roots of large degree, which is characterized in using the two-element public-key. In the construction it is used a prime modulus p having the following structure $p = Nt_1t_2 + 1$, where N is an even number; t_1 and t_2 are prime numbers such that $|t1| \approx |t_2| \geqslant 80$ bits. In contrast to the DS scheme described previously, the public key Y is formed in the form of two numbers, which are calculated using the formulas $Y_1 = K_1^{t_1} K_2^{t_2} \mod p$ and $Y_2 = K_3^{t_1} K_4^{t_2} \mod p$, where four numbers $K_1 < p$, $K_2 < p$, $K_3 < p$, and $K_4 < p$ are the private key. The digital signature is a triple e, S_1, and S_2.

Suppose a message M is given. The signature generation procedure is performed as follows:

1. Select at random two numbers T_1 and T_2.
2. Calculate the value $R = T_1^{t_1} T_2^{t_2} \pmod{p}$.
3. Calculate the first signature element e: $e = F(R, M) = RH \pmod{w_1}$,

where H is the hash value computed from the message: $H = F_H(M)$. The value e is represented as the concatenation of two values e_1 and e_2: $e = e_1||e_2$.

4. Calculate second signature element S_1 using the following formula $S_1 = T_1 K_1^{-e_1} K_3^{-e_2} \pmod{p}$.
5. Calculate the third signature element S_2 using the following formula $S_2 = T_2 K_2^{-e_1} K_4^{-e_2} \pmod{p}$.

The signature verification algorithm is as follows.

1. Using the signature (e, S_1, S_2) calculate the value $R' = Y_1^{e_1} Y_2^{e_2} S_1^{t_1} S_2^{t_2} \pmod{p}$.
2. Calculate the value $e' = F(R', M) = R'H \pmod{t_1}$.
3. Compare e' with e. If $e' = e$, then the signature is valid.

3.2. Proof that signature verification works

If the digital signature has been formed correctly, i.e., using the true private key in accordance with the specified procedure of the signature generation, then in step 3 of the signature verification procedure it is obtained the value e' equal to e. On the basis of the equality $e' = e$ it is concluded about validity of the digital signature. Correctness of the signature scheme can be proved as follows. Substituting into the formula $R' = Y_1^{e_1} Y_2^{e_2} S_1^{t_1} S_2^{t_2} \pmod{p}$ the values $Y_1 = K_1^{t_1} K_2^{t_2} \pmod{p}$, $Y_2 = K_3^{t_1} K_4^{t_2} \pmod{p}$, $S_1 = T_1 K_1^{-e_1} K_3^{-e_2} \pmod{p}$, and $S_2 = T_2 K_2^{-e_1} K_4^{-e_2} \pmod{p}$, we obtain:

$$
\begin{aligned}
&R' = Y_1^{e_1} Y_2^{e_2} S_1^{t_1} S_2^{t_2} \pmod{p} = \\
&(K_1^{t_1} K_2^{t_2})^{e_1} (K_3^{t_1} K_3^{t_2})^{e_2} (T_1 K_1^{-e_1} K_3^{-e_2})^{t_1} (T_2 K_2^{-e_1} K_4^{-e_2})^{t_2} \pmod{p} = \\
&T_1^{t_1} T_2^{t_2} \pmod{p} = R
\end{aligned}
$$

i.e., the value R' obtained at the first step of the signature verification algorithm is equal to R, so $e' = R'H \mod t_1 = RH \mod t_1 = e$.

3.3. Security discussion

The variants of the attack presented in Section 2.3 can be also applied against the second DS scheme. Details of the algorithms for forging the signature are different, but the used ideas and approaches are similar to the case of attacking the first signature scheme. The first two variants of the attack dictate the need of the choice of the size of prime powers t_1 and t_2 equal to $|t_1| = |t_2| \geqslant 80$ bits. The third type attack, based on solving the discrete logarithm problem, determine the size of the prime modulus $|p| = 1024$ bits, which provides 80-bit security of the considered signature scheme.

4. Conclusion

The proposed two constructions of the signature algorithms illustrates two new approaches to design of the digital signature schemes based on the difficulty of finding large prime roots in the ground finite fields. The cryptosystems can be broken with solving the discrete logarithm problem in the finite ground field like in the case of the cryptosystem described in [2]. To obtain the 80-bit security of the cryptosystems based on difficulty of finding roots in the finite field GF(p) one should use the 1024-bit value p. The advantage of the proposed approaches against the construction introduced in [4] consists in possibility to construct fast signature schemes based on difficulty of finding roots in the finite groups of the EC points.

Acknowledgement. The first author was supported by Russian Foundation for Basic Research grant # 11-07-00004-a.

References

[1] **N. Koblitz and A.J. Menezes**, *Another look at "Provable Security"*, J. Cryptology **20** (2007), 3 – 38.

[2] **N.A. Moldovyan**, *Digital signature scheme based on a new hard problem*, Computer Sci. J. Moldova **16** (2008), 163 – 182.

[3] **A.A. Moldovyan, D.N. Moldovyan and L.V. Gortinskaya**, *Cryptoschemes based on new signature formation mechanism*, Computer Sci. J. Moldova **14** (2006), 397 – 411.

[4] **A.J. Menezes and P.C. Van Oorschot**, *Handbook of applied cryptography*, CRC Press, Boca Raton, 1997.

[5] **A.J. Menezes and S.A. Vanstone**, *Elliptic curve cryptosystems and their implementation*, J. Cryptology **6** (1993), 209 – 224.

[6] **R.L. Rivest, A. Shamir and L.M. Adleman**, *A method for obtaining digital signatures and public key cryptosystems*, Commun. ACM **21**, (1978), 120 – 126.

Received April 05, 2012

N.A. Moldovyan
St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences,
14 Liniya, 39, 199178, St. Petersburg, Russia
E-mail: nmold@mail.ru

V.A. Shcherbacov
Institute of Mathematics and Computer Science of the Academy of Sciences, Academiei 5,
MD-2028, Chisinau, Moldova
E-mail: scerb@math.md