Public-key cryptosystem design based on factoring and discrete logarithms

L. Harn

Indexing terms: Public-key cryptosystem, Factoring, Discrete logarithms

Abstract: Most existing cryptosystem designs incorporate just one cryptographic assumption, such as factoring or discrete logarithms. These assumptions appear secure today; but, it is possible that efficient algorithms will be developed in the future to break one or more of these assumptions. It is very unlikely that multiple cryptographic assumptions would simultaneously become easy to solve. Enhancing security is the major objective for cryptosystems based on multiple assumptions. K.S. McCurley proposed the first key distribution system based on two dissimilar assumptions, both of which appear to be hard. In his design, the sizes of the security parameters for these two assumptions are quite different. The modulus to satisfy the proper security requirement for one assumption is too large for the other assumption. The side effects are (1) the public key size is larger than the original Diffie–Hellman key distribution scheme; and (2) more computation time is required. In the paper, the authors propose a cryptographic system design based on the two popular assumptions: factoring and discrete logarithms. Breaking this system is computationally infeasible because it requires (1) solving the Diffie–Hellman discrete logarithm problem in a subgroup of Z_p^*, where $p = 2p' \times q' + 1$ and p', q' are two large primes, and (2) factoring $(p-1)/2$ into two large primes, p' and q'. Thus, in the proposed system, it is possible to choose the same size of security parameter for these two assumptions and, therefore, to maintain the efficiency of the implementation.

1 Introduction

In 1976 Diffie and Hellman [1] proposed the concept of the public-key cryptosystem to solve the secret communication key distribution problem. Since then several public-key cryptosystems [2–6] which can provide both digital signature and encryption have been proposed. One common feature among all these systems is that the security of each cryptosystem is based on just one cryptographic assumption, such as factoring or discrete logarithms. According to Reference 7, the solution of the discrete logarithm requires $O\{\exp[\text{const.}\sqrt{(\log p \log \log p)}]\}$ integer multiplication, where p is the modulus. For further information on the current state of the art in computing discrete logarithms, see References 8 and 9. According to Reference 10, the asymptotic running times for many integer factorisation algorithms are given in the form of $O\{\exp[\text{const.}\sqrt{(\log n \log \log n)}]\}$, where n is the product of two large primes. For more information on factoring, see References 11 and 12. Recent advanced techniques imply that the computational difficulties of these two assumptions are almost the same. Thus, in order to achieve the same security level for these two different assumptions, the size of the modulus p for the discrete logarithm problem and the size of modulus n for the factoring problem should be the same.

Although these cryptographic assumptions appear secure today, it is still very likely in the future that a clever cryptanalyst will discover an efficient way to factor integers or to compute discrete logarithms. Thus, cryptosystems based on the corresponding assumption will surrender their security. To enhance security is the major motivation for developing cryptosystems based on multiple cryptographic assumptions. This is because of the common belief that it is very unlikely that multiple cryptographic assumptions would simultaneously become easy to solve.

In 1988 K.S. McCurley [13] proposed the first key distribution system based on two dissimilar assumptions, both of which appear to be hard. Instead of using an arithmetic modulus p that is a prime (as in the Diffie–Hellman key distribution scheme), the distribution scheme in Reference 13 uses a modulus n that is a product of two primes. Breaking the system requires the factoring of n into two primes, p and q, and the ability to solve the Diffie–Hellman discrete logarithm problem in subgroups of Z_p^* and Z_q^*. Thus, it is impossible to select proper moduli p and q to achieve the same difficulty for these two assumptions. For example, if we select two large primes p and q with 512 bits each, in order to ensure the difficulty of the discrete logarithm problem in subgroups of Z_p^* and Z_q^*, the resulting composite modulus $n = p \times q$ of the factoring problem will become 1024 bits long. This results in two disadvantages: (1) the public-key size will become two times longer than that of the original Diffie–Hellman scheme; (2) the time to compute a 1024-bit exponentiation is almost eight times longer than the time to compute a 512-bit exponentiation. In 1992 E.F. Brickell and K.S. McCurley [14] proposed an interactive identification scheme also based on discrete logarithms and factoring, but these two assumptions are not as general as the two assumptions stated previously.

The object of this paper is to develop a cryptosystem based on two different cryptographic assumptions to enhance the security, while maintaining the efficiency of

Paper 1040E (C3), first received 20th May and in revised form 21st December 1993

The author is with the Computer Science Telecommunications Program, University of Missouri — Kansas City, Kansas City, MO 64110, USA

the implementation. In other words, one must break the RSA and the ElGamal systems simultaneously to break our proposed system. Computational time in the proposed system for encryption /decryption /signature generation /signature verification is bounded above by the worst case of the RSA and the ElGamal systems (i.e $T = \max\{T_{EIG}, T_{RSA}\}$, where T_{EIG}, T_{RSA}, are the computational times for the ElGamal scheme and RSA scheme, respectively). In Section 2, we propose a public-key distribution system based on these two assumptions. Public-key encryption scheme and digital signature scheme are included in Section 3.

2 Public-key distribution system based on factoring and discrete logarithms

Each user selects a large prime $p = 2p' \times q' + 1$, where $p' = 2p'' + 1$, $q' = 2q'' + 1$, and p', q', p'' and q'' are also large primes, and randomly selects a primitive element $\alpha \bmod p$ and $x \in [1, p-1]$. Then the user computes d such that $3 \times d \bmod \phi(\phi(p)) = 1$, where $\phi(\cdot)$ is the Euler totient function, and $y = \alpha^x \bmod p$. $(p, \alpha, y, 3)$ are the public keys (p', q', x, d) are the secret keys. Note that we use the exponent 3 as the public key to simplify the encryption operation (but not decryption operation). Some precautions need to be taken in order to prevent flaws [15]. All these public and secret keys will be used for the design of the entire cryptosystem.

Suppose A wants to share a common secret key K_{AB} with B during a communication session, where A is the initiator. A first obtains B's public keys (p_B, α_B, y_B), and B has its own secret keys (p'_B, q'_B, x_B, d_B). Then A randomly selects a secret key $k \in [1, p_B - 1]$ and computes K_{AB} as

$$K_{AB} = y_B^k \bmod p_B$$

Furthermore, A computes

$$z_A = \alpha_B^k \bmod p_B$$

and

$$C = z_A^3 \bmod (p_B - 1)$$

and sends C to B. Once B receives C, B uses its secrete key d_B to compute

$$z_A = C^{d_B} \bmod (p_B - 1)$$

and uses its secret key x_B to obtain K_{AB} as

$$K_{AB} = z_A^{x_B} \bmod p_B$$

2.1 Discussion

The security of this proposed scheme is based on factoring and discrete logarithms. An attacker may access B's public key y_B, but knowing how to solve discrete logarithms can only help the attacker to obtain B's secret key x_B. In order to obtain the common session key K_{AB} the attacker also needs to solve the factoring problem to obtain B's secret key d_B. On the other hand, the attacker needs to solve the factoring problem first to obtain z_A and then to solve the discrete logarithm problem to obtain k. Breaking our system requires solving the Diffie–Hellman discrete logarithm problem in a subgroup of Z_p^*, where $p = 2p' \times q' + 1$ and p', q' are two large primes, and the ability to factor $(p-1)/2$ into two large primes, p' and q'. Thus, in our proposed system, we can maintain the same security level for these two cryptographic assumptions. For example, if we require that p', $q' \geq 2^{256}$, then the modulus for both discrete logarithms and factoring problems will be at least 512-bit long. The computational time for A and B requires almost two 512-bit exponentiations each. This is four times faster than the McCurley scheme.

3 Public-key cryptosystem and signature scheme based on factoring and discrete logarithms

3.1 Cryptosystem

3.3.1 Phase 1: Public–key distribution. Suppose A wants to send some secret information to B. First, A obtains and authenticates B's public keys (p_B, α_B, y_B). Then A randomly selects a number k from $[1, p_B - 1]$ with $\gcd(k, \phi(p_B)) = 1$. According to the public-key distribution scheme described previously, a common secret key K_{AB} can be obtained by A as

$$K_{AB} = y_B^k \bmod p_B$$

Note here that this common secret key K_{AB} shared between A and B is also a primitive element mod p_B according to the following corollary based on the K. H. Rosen theorem 8.4 [16].

Corollary 1: K_{AB} is a primitive element mod p_B.

k will serve as A's 'secret session key', and the corresponding K_{AB} will become the 'common secret session key' shared by A and B. Then A computes

$$z_A = \alpha_B^k \bmod p_B$$

and

$$v = y_A^3 \bmod (p_B - 1)$$

and sends v to B.

3.1.2 Phase 2: Encryption. For each message block m_i in a sequence of message blocks $\{m_1, m_2, \ldots, m_i, \ldots\}$, A computes two encryption keys $K_{i,1}$ and $K_{i,2}$ iteratively as

$$K_{i,1} = K_{i-1,1}\, K_{AB} \bmod p_B = K_{AB}^i \bmod p_B$$

and

$$K_{i,2} = \alpha_B^{K_{i,1}} \bmod p_B$$

where $K_{0,1} = 1$.

The corresponding ciphertext block C_i is computed as

$$C'_i = m_i^3 \bmod p_B - 1$$

and

$$C_i = K_{i,2}\, C'_i \bmod p_B$$

The sequence of ciphertext blocks $\{C_1, C_2, \ldots, C_i, \ldots\}$ is transmitted to B.

3.1.3 Phase 3: Decryption: Once B receives v from A, B can use its secret key d_B to compute z_B as

$$z_A = v^{d_B} \bmod (p_B - 1)$$

and use its secret key x_B to obtain the common secret session key K_{AB} as

$$K_{AB} = z_A^{x_B} \bmod p_B$$

and encryption keys $K_{i,1}$ and $K_{i,2}$ as well. For each received ciphertext C_i the corresponding message m_i is computed as

$$C'_i = C_i K_{i,2}^{-1} \bmod p_B$$

$$m_i = C_i'^{d_B} \bmod (p_B - 1)$$

where $K_{i,2}^{-1}$ is the multiplicative inverse of $K_{i,2} \bmod p_B$ and $3 \times d_B \bmod \phi(\phi p_B)) = 1$. But, according to Fermat's theorem [17, pp. 42], we have

$$K_{i,2}^{-1} = \alpha_B p_B^{-1-K_{i,1}} \bmod p_B$$

and $K_{i,2}^{-1}$ can be computed without knowing $K_{i,2}$, thus speeding up the computation.

3.1.4 Discussion

(1) In our proposed cryptosystem, one modular exponentiation is required for enciphering each message block (the time required for modular exponentiation with exponent 3 can be ignored) and two modular exponentiations are required for deciphering each ciphertext block. This performance is very similar to the original ElGamal scheme. However, the ElGamal scheme requires two modular exponentiations for enciphering one message block and one modular exponentiation for deciphering one cipher text block.

(2) In our proposed cryptosystem, for every message block m_i, there is a corresponding ciphertext block C_i. The transmission efficiency is 1:1.

(3) In our system, the unique secret session key k chosen by A, and the unique common secret session key K_{AB} shared by A and B, are used throughout the session to generate the encryption keys $K_{i,1}$ and $K_{i,2}$ for $i = 1, 2, \ldots$. These keys K_{AB}, $K_{i,1}$ and $K_{i,2}$ will differ from one session to the next if the value of k is different. Hence, k should be selected randomly to ensure security. Note that since K_{AB} is a primitive element of $GF(p_B)$, we can obtain a period of length $p_B - 1$ for the encryption keys $K_{i,1}$ and $K_{i,2}$. So, for all practical applications, the period length is as long as we want. In other words, within a session, even though we use the same k to generate the encryption keys $K_{i,1}$ and $K_{i,2}$, these keys differ for each message block.

3.2 Digital signature

Suppose A wants to sign a message m, where $0 \leqslant m \leqslant p_A - 1$. A randomly selects a number k from $[0, p_A - 1]$ with $\gcd(k, \phi(p_A)) = 1$ and computes

$$r = \alpha_A^k \bmod p_A$$

A now solves the congruence

$$m' = ks' + x_A r \bmod p_A - 1$$

for the integer s' and computes

$$s = s'^{d_A} \bmod p_A - 1$$

The signature for message m is then the ordered pair (r, s).

Upon receiving the set of $\{m, r, s\}$, any user can verify the signature of message m by computing

$$s' = s^3 \bmod p_A - 1$$

and checking the following equation:

$$\alpha_A^{m'} = r^{s'} y_A^r \bmod p_A$$

3.2.1 Discussion Since r and $k^{-1} \bmod p_A - 1$ can be precomputed offline, the computational time for a designer is almost the same as for the RSA scheme. On the other hand, since the computational time required for modular exponentiation with exponent 3 can be ignored, the computational time for a verifier is almost the same as for the original ElGamal scheme. Thus, the computational time of our proposed scheme is upper bounded by the worst case of the RSA and ElGamal schemes.

4 Conclusions

We have proposed a public-key cryptosystem design based on factoring and discrete logarithms, to enhance the security, while maintaining the efficiency of the implementation. One must break the RSA and the ElGamal systems simultaneously to break our proposed system.

There are several open problems. Is there any other approach that offers the same security as we propose but with better performance? Can we design a cryptosystem based on other multiple assumptions but with better performance? Instead of letting each user select his own p, p' and q', is it possible to allow a trusted key centre to select these parameters?

5 References

1 DIFFIE, W., and HELLMAN, M.E.: 'New directions in cryptography', *IEEE Trans.*, 1976, **IT-22**, pp. 644–654
2 RIVEST, R.L., SHAMIR, A., and ADELMAN, L.: 'A method for obtaining digital signatures and public-key cryptosystem', *Commun. ACM*, 1978, **21**, (2), pp. 120–126
3 ELGAMEL, T.: 'A public key cryptosystem and a signature scheme based on discrete logarithms', *IEEE Trans.*, 1985, **IT-31**, pp. 469–472
4 RABIN, M.O.: 'Digitalized signatures and public key functions as intractable as factorization', MIT/LCS/TR-212, January, 1979
5 McELIECE, R.J.: 'A public-key cryptosystem based on algebraic coding theory', DSN Progress Report, 42–44, pp. 114–116, 1978
6 KOYAMA, K., MAURER, U., OKAMOTO, T., and VANSTONE, S.A.: 'New public-key schemes based on elliptic curves over the ring Z_n'. Advances in Cryptology — CRYPTO '91 (Springer, Berlin, 1992)
7 ODLYZKO, A.M.: 'Discrete logarithms in finite fields and their cryptographic significance'. Advances in Cryptology — EUROCRYPT '89 (Springer, Berlin, 1990), pp. 224–314
8 LAMACCHIA, B., and ODLYZKO, A.: 'Computation of discrete locarithms in prime finite fields'. Advances in Cryptology — CRYPTO '90 (Springer, Berlin, 1991)
9 McCURLEY, K.S.: 'The discrete logarithm problem'. Proceedings of Symposia in *Applied Mathematics*, Providence, Rhode Island, 1990, Vol. 42, American Mathematical Society, pp. 49–74
10 POMERANCE, C.: 'Analysis and comparison of some integer factoring algorithms', *Computational Methods in Number Theory*, 1982, **154**, pp. 89–139
11 LENSTRA, A.K., and MANASSE, M.S.: 'Factoring by electronic mail'. Advances in Cryptology — EUROCRYPT '89 (Springer, Berlin, 1990), pp. 355–371
12 POMERANCE, C.: 'Factoring'. Proceedings of Symposia in *Applied Mathematics*, Providence, Rhode Island, 1990, Vol. 42, American Mathematical Society, pp. 27–48
13 McCURLEY, K.S.: 'A key distribution system equivalent to factoring', *J. Cryptology*, 1988, **1**, (2), pp. 95–106
14 BRICKELL, E.F., and McCURLEY, K.S.: 'An interactive identification scheme based on discrete logarithms and factoring', *J. Cryptology*, 1992, **5**, (1), pp. 29–40
15 HASTAD, J.: 'Solving simultaneous modular equations of low degree', *SIAM J. Comput.*, 1988, **17**, (2), pp. 336–341
16 ROSEN, K.H.: 'Elementary number theory and its applications' (Addison-Wesley, 1992, 3rd edn)
17 DENNING, D.E.R.: 'Cryptography and data security' (Addison-Wesley, 1982)