A New Public Key Cryptosystem based on Matrices

RAFAEL ALVAREZ[1], FRANCISCO-MIGUEL MARTINEZ[2], JOSE-FRANCISCO VICENT[3], and ANTONIO ZAMORA[4]
Departamento de Ciencia de la Computación e Inteligencia Artificial
Universidad de Alicante
Campus de Sant Vicent del Raspeig, Ap. Correos 99, E-03080, Alicante
SPAIN

This work was partially supported by the Spanish grants GV06/018

Abstract: - This paper describes a new method for authentication and integrity where the ciphertext is obtained using block upper triangular matrices with elements in $\mathbb{Z}_p$, in which the discrete logarithm problem (DLP) defined over a finite group is used. In the proposed public key cryptosystem, the encryption requires very few operations and decryption is equivalent to the DLP and, finally, the signature scheme presented is based on the ElGamal signature scheme and requires the original message in order to verify the signature. With this system we get a large key space without increasing the difficulty of the problem.

Keywords: - Cryptography, Security, Public-Key, DLP, Finite Fields, Diffie-Hellman, Polynomial Matrices, ElGamal, Digital Signature.

1 Introduction

In order to establish a confidential channel between two users of a network, classical single-key cryptography requires them to exchange a common secret key over a secure channel. This may work if the network is small and local, but it is infeasible in non-local or large networks.

To simplify the key exchange problem, modern public-key cryptography provides a mechanism in which the keys to be exchanged do not need to be secret. In such a framework, every user possesses a key pair consisting of a (non-secret) public key and a (secret) private key; only public keys are published.

They are used to encrypt the messages to be sent to the owner of the key or to verify digital signatures issued by the owner of the key. Before using someone else's public key to encrypt a message or verify a signature, one should make sure that the key really belongs to the intended recipient or the indicated issuer of the signature.

Achieving authenticity of public keys can be done in several ways. Public key cryptosystems are essential for electronic commerce or electronic banking transactions; they assure privacy as well as integrity of the transactions between two parties. Digital signatures are used to sign electronic documents and they are also mostly based on public-key techniques.

A lot of popular public-key encryption systems are based on number-theoretic problems such as factoring integers or finding discrete logarithms. The underlying algebraic structures are, very often, abelian groups; this is especially true in the case of the Diffie-Hellman method (DH, see [5]), that was the first practical public key technique and introduced in 1976.

The Discrete Logarithm Problem (DLP, see [4, 11, 12]) is, together with the Integer Factoring Problem (IFP) and the Elliptic Curve DLP (ECDLP, see [2]), one of the main problems upon which public-key cryptosystems are built. Thus, efficiently computable groups where the DLP is hard to break are very important in cryptography. In recent years, cryptographic research has become more and more important due to the increasing number of application areas related to the field, requiring data confidentiality, authentication and integrity.

The method presented in this paper, generalises the DH approach to a group based on the powers of a block upper triangular matrix, which is a very flexible and practical technique.

The usual sizes for the keys in the IFP or DLP are around 1024 binary digits, existing well known algorithms of sub-exponential order that solve these problems (see [7, 9, 10]).

The so called square root algorithms (see [8, 13, 14, 16]) reach an order of complexity $\sqrt{p}$ where p is the greater prime factor of the order of the group. This is not enough to be used in big and arbitrary finite groups, but if this order does not have great prime factors, these algorithms can be

practical. Therefore it is necessary that the order of the group in which we are working has great prime factors.

Our system is capable of increasing the computational cost required for a successful attack on the generated DLP for equivalent key sizes.

The rest of the paper is divided as follows: section 2 shows some properties necessary for the proposed cryptosystem. Section 3 is divided in several subsections: a key exchange protocol, an encryption scheme and a digital signature scheme. Finally, several conclusions about the system are given in section 4.

2 Preliminaries

Some basic linear algebra properties, necessary for the purpose of the paper, are presented in this section; for a more in depth treatment see [1, 3].

Given p a prime number and $r, s \in \mathbb{Z}$, we denote by $Mat_{r\times s}(\mathbb{Z}_p)$ the matrices of size $r\times s$, with elements in $\mathbb{Z}_p$, and by $GL_r(\mathbb{Z}_p)$ and $GL_s(\mathbb{Z}_p)$ the invertible matrices of size $r\times r$ and $s\times s$.

We define

$$\theta=\left\{\begin{bmatrix} A & X \\ \mathbf{0} & B \end{bmatrix}, A\in GL_r(\mathbb{Z}_p), B\in GL_s(\mathbb{Z}_p), X\in Mat_{r\times s}(\mathbb{Z}_p)\right\}.$$

Theorem 1 *The set θ has a structure of a non abelian group for the product of matrices.*

Theorem 2 Let $M=\begin{bmatrix} A & X \\ \mathbf{0} & B \end{bmatrix}\in\theta$, we *consider the subgroup generated by the different powers of M.*

Taking h as a non negative integer then

$$M^h=\begin{bmatrix} A^h & X^{(h)} \\ \mathbf{0} & B^h \end{bmatrix}, \quad (1)$$

where

$$X^{(h)}=\begin{cases} \mathbf{0} & if \quad h=0, \\ \sum_{i=1}^{h} A^{h-i}XB^{i-1} & if \quad h\geq 1. \end{cases} \quad (2)$$

Also, if $0\leq t\leq h$, then

$$X^{(h)}=A^tX^{(h-t)}+X^{(t)}B^{h-t}, \quad (3)$$
$$X^{(h)}=A^{h-t}X^{(h)}+X^{(h-t)}B^t. \quad (4)$$

As a consequence, in the case $t = 1$ we have

$$X^{(h)}=AX^{(h-1)}+XB^{h-1},$$

$$X^{(h)}=A^{h-1}X+X^{(h-1)}B,$$

and, taking a, b integers such as $a+b\geq 0$, we have

$$X^{(a+b)}=A^aX^{(b)}+X^{(a)}B^b. \quad (5)$$

In this scheme, the key space is bound to the order of the group generated by the M matrices. For this reason, we present next the way to guarantee that this order is sufficiently high.

Let $f(x)=a_0+a_1x+\cdots+a_{r-1}x^{r-1}+x^r$ a monic polynomial in $\mathbb{Z}_p[x]$ and

$$\overline{A}=\begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix}$$

its companion matrix. If $f(x)$ is a primitive polynomial then the order of $\overline{A}$ is exactly p^n-1. Consequently, if we work in $\mathbb{Z}_p[x]$, it is possible to easily construct matrices whose order is maximal.

Constructing matrix M using primitive polynomials we can guarantee a certain order.

Let

$$f(x)=a_0+a_1x+a_2x^2+\cdots+a_{r-1}x^{r-1}+x^r,$$
$$g(x)=b_0+b_1x+b_2x^2+\cdots+b_{s-1}x^{s-1}+x^s,$$

be two primitive polynomials in $\mathbb{Z}_p[x]$, and $\overline{A}, \overline{B}$ the corresponding associated matrices; let P, Q be two invertible matrices, $A=P\overline{A}P^{-1}$ and $B=Q\overline{B}Q^{-1}$.

With this construction, the order of M is $o(M)=lcm(p^r-1,\ p^s-1)$; this number will be maximal if we take r and s prime.

In table 1, where the value that appears in the column *o(M)* represents the number of decimal digits (the integer 2^{128} has 39 digits), it can be observed that the values of r and s do not need to be very big to optimize the order.

It is easy to reduce a generic DLP in a cyclic group (with order $o(M)$) whose factorization is known. It is very important in the election of the group that the order is prime or at least with very big prime factors. So if $o(M)$ is a prime number, it will

require on the order of $\sqrt{m}$ operations to compute the discrete logarithm in group θ.

Table 1. Order of M, for different values of p, r and s

p	*r*	*s*	*o(M)*	*p*	*r*	*s*	*o(M)*
3	32	31	30	29	31	32	82
	48	47	39		47	48	97
	64	63	47		60	61	103
	130	131	145		130	131	311
5	32	31	38	31	16	15	40
	30	33	39		32	31	87
	64	63	61		64	63	111
	130	131	184		131	131	342
7	24	27	39	251	12	13	46
	32	31	43		32	31	276
	64	63	70		64	63	457
	130	131	213		130	131	1379
11	22	21	39	257	9	10	40
	32	31	50		32	31	287
	64	63	77		64	63	479
	130	131	239		130	131	1479

3 The algorithms

3.1 Key exchange protocol

We will see now the proposed system of block matrices applied to the DH key exchange protocol.

Let *U* and *V* be two interlocutors who wish to exchange a key, then

1. *U* and *V* agree on $p \in \mathbb{Z}$,
   $M_1 = \begin{bmatrix} A_1 & X_1 \\ \mathbf{0} & B_1 \end{bmatrix} \in \theta$, with order m_1
   and
   $M_2 = \begin{bmatrix} A_2 & X_2 \\ \mathbf{0} & B_2 \end{bmatrix} \in \theta$, with order m_2.
2. *U* randomly generates two private keys r, s with $1 \le r \le m_1$, $1 \le s \le m_2$, computes $C = M_1^{\,r} M_2^{\,s}$ and publishes this value.
3. *V* randomly generates two private keys v, w with $1 \le v \le m_1$, $1 \le w \le m_2$, computes
   $$\begin{aligned} F &= M_1^{\,v} M_2^{\,w}, \\ D &= M_1^{\,v} C M_2^{\,w} \\ &= M_1^{\,v} M_1^{\,r} M_2^{\,s} M_2^{\,w} \\ &= M_1^{\,v+r} M_2^{\,s+w} \\ &= M_1^{\,r+v} M_2^{\,w+s} \\ &= M_1^{\,r} M_1^{\,v} M_2^{\,w} M_2^{\,s}, \end{aligned}$$
   and publishes this matrix.
4. *U* calculates $M_1^{-r} M_2^{-s}$ and
   $$\begin{aligned} F &= M_1^{-r} D M_2^{-s} \\ &= M_1^{-r} M_1^{\,r} M_1^{\,v} M_2^{\,w} M_2^{\,s} M_2^{-s} \\ &= M_1^{\,v} M_2^{\,w}. \end{aligned}$$
5. The public key of *U* and *V* are respectively *C* and *D*.

In this way, the key shared by *U* and *V* is *F*, now both interlocutors, share a common and secret element.

An attacker could know *p* and *M*, but to obtain the shared secret would have to face a problem with a complexity similar to that of the DLP (see [4]).

3.2 Data encryption

We have to start from the same public and private elements seen previously in the key exchange protocol (which we suppose already done).

The interlocutor *U* wishes to, privately, send a message to *V*. The message must be coded as a matrix $\Delta \in Mat_{r \times s}(\mathbb{Z}_p)$.

Encryption:

1. U builds the matrices
   $T = \begin{bmatrix} A_1 & \Delta \\ \mathbf{0} & B_1 \end{bmatrix}$ and F,
   that are invertible since A_1, A_2, B_1 and B_2 are invertible too.
2. *U* computes matrix $C = TF$ and sends this matrix to *V*.

Decryption:

1. *V* computes the inverse of the matrix *F*.
2. *V obtains* T carrying out the product CF^{-1}.
3. *V* recovers the message Δ selecting, the respective block of T.

With this, the functions of encryption and decryption of the interlocutor *V* would be respectively

1. $E_{k_2}(\Delta) = TF$.
2. $D_{k_2}(C) = CF^{-1} = T$.

With the appropriate quick exponentiation algorithms (see [4]), the powers of the matrices can be computed efficiently.

The complexity of the problem that an attacker would face is in the order of that of the DLP, acting, in effect, as a deterrent for a possible attack.

3.3 Signature scheme

We propose a digital signature scheme that requires the original message in order to verify the signature.

The scheme, that follows, is based on the ElGamal (see [6]) digital signature scheme.

We suppose that the users *U* and *V* have exchanged the key *F*, and *U* has sent the message Δ to *V*, according to the previous protocol. If the transmitter *U* wishes to digitally sign the message Δ proceeds in the following way

1. *U* generates a random number *r*.
2. *U* computes F^r.
3. With *T* computes $Q = T - F^r$.
4. The digital signature is (r, Q).

If the receiver wishes to verify the digital signature of *U*, he proceeds in the following way

1. V computes F^r and then $Q + F^r = T$
2. *V* extracts de corresponding block of *T* named *Y* and compares Δ and *Y*, turning out to be an authentic signature if $\Delta = Y$ and false if $\Delta \neq Y$.

4 Conclusions

We have presented a public key cryptosystem based on a generalization of the DLP for block matrices with elements in $\mathbb{Z}_p$, with the advantage of reducing the required key length for a given level of security. This cryptosystem provides an efficient protection against common attacks without the need of bigger key sizes.

We have defined a set of matrices θ constructed using primitive polynomials. Due to this we can work with big groups, without the need to use neither enormous matrices nor high numbers.

Given two parties, the key exchange protocol guarantees that both parties share a secret element of set *G*; the public key cryptosystem defined assures data confidentiality and the digital signature scheme guarantees authentication and integrity.

References:

[1] Alvarez, R., Ferrández, F., Vicent, J-F., Zamora, A. Applying quick exponentiation for block upper triangular matrices. Applied Mathematics and Computation, 183-2, 729-737. 2006.

[2] Blake, I., Seroussi, G. Smart, N. Elliptic Curves in Cryptography. London Mathematical Society Lecture Notes Series 265. Cambridge University Press. 1999.

[3] Climent, J-J., Ferrández, F., Vicent, J-F., Zamora, A. A non-linear elliptic curve cryptosystem based on matrices. Applied Mathematics and Computation, Volume 174-1, 150-164. 2006.

[4] Coppersmith, D., Odlyzko, A., and Schroeppel, R. Discrete logarithms in GF(p). Algorithmica 1-15. 1986.

[5] Diffie, W., Hellman, M. New directions In Cryptography. IEEE Trans. Information Theory. 22: 644-654. 1976.

[6] Elgamal, T. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE Trans. Inform. Theory. 31: 469-472. 1985.

[7] Gordon, D. M. A Survey of Fast Exponentiation Methods. Journal of Algorithms. 27: 129–146. 1998.

[8] Hellman, M.E., Reyneri, J.M. Fast computation of discrete logarithm in GF(p). Advances in cryptology: Proceedings of GRYPTO'82 Plenum Press. 3-13. 1983.

[9] Hoffman, K., Kunze, R. Linear Algebra. Prentice-Hall. New Jersey. 1971.

[10] Koblitz, N. A Course in Number Theory and Cryptography. Springer-Verlag. 1987.

[11] McCurley K. The discret logarithm problem. Crytology and Computational Number Theory, Proceedings of Symposia in Applied Mathematics. 42: 49–74. 1990.

[12] Menezes, A., Wu, Y-H. The Discrete Logarithm Problem in GL(n,q). Ars Combinatoria. 47: 22-32. 1997.

[13] Pohlig. S, Hellman, M. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Trans. 24: 106-110. 1979.

[14] Pollard, J.M. Monte Carlo methods for index computation (mod p). Math. Computation. 32: 918-924. 1978.

[15] Rivest, R., Shamir, A., Adleman, L. A Method for Obtaining Digital Signatures and Public Key Cryptosystems. ACM Communications. 21: 120-126. 1978.

[16] Shanks, D. Class number, a theorey of factorization and generation. Number Theory Institute. Proc. Symposium pure Mathematics. American Mathematics Society. 20: 415-440. 1981.