An Application of Algebraic Curves for Key Exchange on Mobile Grid Computing

I. JIRÓN [1] I. DERPICH[1] I. SOTO [1] R. CARRASCO [2]
[1] Industrial Engineering Department
Engineering Faculty
University of Santiago of Chile
Av. Ecuador 3769, Santiago.
CHILE

[2] School of Electrical, Electronic and
Computer Engineering.
University of Newcastle upon Tyne
Merz Court Newcastle upon Tyne (NE1 7RU)
UNITED KINGDOM

Abstract: The aim of this paper is to show a new algorithm for key exchange and reliable data transmission through the mobile grid computing environment. The system uses very strong one way functions based on the combination of a family of algebraic curves and a Low-Density Parity-Check code (LDPC). The security of the whole scheme is guaranteed by the intractability of the discrete logarithm problem on several algebraic curves. Furthermore, the algorithm complexity is studied. Finally, the proposed system produces an improvement of 0.82 dB at a BER = 10^{-4} which is achieved when it is compared with the LDPC code over Rayleigh Fading channel with BPSK modulation.

Key-Words Mobile Grid-Computing, Low-Density Parity-Check code, Algebraic Curves, BPSK modulation.

1. Introduction

With the recent advances in Grid computing and service-oriented architectures, computing is becoming increasingly less confined to the traditional computing platforms of desktops, servers or mainframes. While Grid computing itself promises the accessibility of vast computing and data resources across geographically dispersed areas, there is currently a lack of established support for Grid-based mobile computing. Grid-enabled computing with mobile devices can be very effective in a multitude of business environments. Users can have access to the computing power and data repositories on the Grid while working out on the field. Mobile Grid access also enables and encourages distributed, collaborative problem-solving environments [1][2][3].

The security in Grid computing could be a showstopper. A user doesn't want a bad remote application browsing your hard drive for confidential information. Security issues include secure sign-on, authentication, authorization, access rights, and privileges. Reliable and secure communications are obtained using cryptography. Protections must be in place to prevent passive intrusion, intercepted data transfers and transactions, tampering, and network disruptions. Maintaining confidentiality and privacy will also be issues if you are transferring personal data [4].

The main objectives of the cryptography are the information confidentiality, data integrity, user authentication, signature and non-repudiation [5]. A traditional way to protect e-commerce has been implemented using cryptosystems for public key cryptography as Diffie-Hellman (DH) and Rivest, Shamir, and Adleman (RSA), invented in 1976 and 1978, respectively.

The Algebraic curves can be used in Cryptography and these are an alternative for cryptosystem based on integer factorization problem as RSA. In 1989, Koblitz [6] recommended the use of the Jacobian group of hyperelliptic curves (HC) for cryptography to provide a large class of algebraic curves. For general HC of small genus the discrete logarithm problem (DLP) in the Jacobian of a HC is thought to be difficult. To ensure the security of a cryptosystem based on the DLP in a finite Abelian group, it is important that the order should either be large prime or the group order contains a large prime factor. A mature revision of such topics can be found in [7]. Recent works have shown that the HC offer smaller size operands compared with Elliptic curves (EC), since the underlying field for a HC is smaller. Therefore, if an 80-bit security level is desired, the underlying field should have an order of approximately 2^{160} with genus 1 and 2^{80} with genus two [8]. The HC also offer faster arithmetic than EC or at least achieves the same computational time for the same security level [8] [9].

Another relevant part of this scheme is a LDPC code. These codes are a class of linear block codes. The name comes from the characteristic of their parity-check matrix which contains only a few 1's in comparison to the amount of 0's with dimension $N \times K$. Their main advantage is that they provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding. Furthermore are they suited for implementations that make heavy use of parallelism. They were first introduced by Gallager in his PhD thesis in 1960 [10]. But due to the computational effort in implementing coder and encoder for such codes and the introduction of Reed-Solomon codes [11], they were mostly ignored until about ten years ago. It allows the coding and decoding of information in order to transmit it. It also has the capacity to detect and correct errors [12].

The main contribution of this paper is to show a simple key exchange algorithm for mobile grid computing environment using the concatenation of a secret family of HC with a LDPC code.

The remainder of this paper will be organized as follows. In section 2 the system description is made. In section 3 the results are presented. Finally, conclusions are presented in section 4.

2. Problem Solution

A particular kind of algebraic curve is used in the proposed system and these curves are classified by the genus [13]. In topology, the genus of a surface is an integer representing the maximum number of cuts that can be made through it without separating it. This is roughly equivalent to the number of holes in it, or handles on it.

A hyperelliptic curve C is an algebraic curve of genus $\boldsymbol{g} \geq 1$ over a Galois field $F = GF(2^{\boldsymbol{n}})$ is formed by the points $P = (u, v) \in \overline{F} \times \overline{F}$, which satisfy the Equation (1), with a point ∞ at infinity [7]:

$$C: v^2 + h(u)v = f(u) \qquad (1)$$

where h and f are polynomials with $\deg(h(u)) \leq \boldsymbol{g}$, $\deg(f(u)) = 2\boldsymbol{g}+1$ and $f(u)$ is monic. A semireduced divisor is defined as:

$$D = \sum m_i P_i - \left(\sum m_i\right)\infty \qquad (2)$$

where P_i is a point on C and $m_i \in \boldsymbol{Z}$, $m_i \geq 0$. Additionally, a reduced divisor D has an order $\sum m_i \leq \boldsymbol{g}$ on C. A reduced divisor only belongs to a unique equivalence class in the Jacobian of C. The Jacobian is the quotient group $\boldsymbol{J}(C) = \boldsymbol{D}^0 / \boldsymbol{P}$, where $\boldsymbol{D}^0$ is the group of divisors of order zero on C and $\boldsymbol{P}$ is the group of the principal divisors on C. $\boldsymbol{P}$ is a subgroup of $\boldsymbol{D}^0$. Since HC of genus $\boldsymbol{g} = \boldsymbol{2}$ are considered in this paper, then a reduced divisor D is represented by polynomial form $D = div(a(u), b(u))$, where $a(u) = \alpha^{e_0} + \alpha^{e_1}u + \alpha^0 u^2$ and $b(u) = \alpha^{s_0} + \alpha^{s_1}u$ belong to $GF(2^{\boldsymbol{n}})[u]$ [7].

Figure 1 depicts an organization of a mobile computational grid. Here, a Middleware application running on a Grid Computing Resource Broker (GCRB) manages a small set of processors and an integrated data store, which are provided by the mobile devices called Users. The Middleware and GCRB are implemented in the base station (BS) of mobile scheme. In this grid environment a User can be client when this makes a Job Request or can be server when this develops the necessary work for satisfies a Job Request. For example, the client User 1 makes a Job Request 1 to Middleware and it sends this to GCRB which makes a routing job towards servers User 2, User 3, …, User $\boldsymbol{n}$. In this case, workload can be broken up and sent in manageable pieces to idle server cycles. Next, the servers return the respectively result pieces to GCRB and Middleware which arm the Result 1 and they give it to the User 1.

Algorithm 1 explains the multifunctional key exchange process and it summarizes the complete grid identity process in which the User 1 and the BS establish a collection of keys, for identification of User 1 with the rest of the users. Then, the resources in servers User 2, User 3, …, User $\boldsymbol{n}$ can be used by User 1 through the mobile grid computing environment.

Algorithm 1: Multifunctional Key Exchange Algorithm.

1. User 1 and BS choose a genus two HC suitable for cryptography C over a $GF(2^{\boldsymbol{n}})$, which is public. Then, User 1 and BS apply the Diffie-Hellman key exchange and the common secret key $k_t(k_r D_l) = k_r(k_t D_l) = div(a^{ck}(u), b^{ck}(u))$ is generated.

2. The following array $\Omega = \left\lfloor a_{i_\gamma}\ a_{i_{\gamma-1}} \ldots a_{i_1}\ b_{j_\delta}\ b_{j_{\delta-1}} \ldots b_{j_1} \right\rfloor = \left[\ \omega_{\gamma+\delta} \ldots \omega_2\ \omega_1\ \right]$ is constructed, which contains all nonzero coefficients of $a^{ck}(u)$ and $b^{ck}(u)$ respectively. Where, $\gamma = \#\left\{ a_i \in GF(2^n) \,/\, a_i \neq 0 \right\}$ and $\delta = \#\left\{ b_j \in GF(2^n) \,/\, b_j \neq 0 \right\}$.

3. User 1 and BS construct the secret HC family $\{C_\tau\}_{\tau=1}^{Per}$:

$$C_\tau : v^2 + uv = u^5 + f_3 u^3 + u^2 + f_0$$

over $GF(2^n)$, where the coefficients f_3 and f_0 are selected according to look-up table1. In this table each row is a 2-permutation of components in the array Ω and there are

$$Per = \binom{\gamma+\delta}{2} = \frac{(\gamma+\delta)!}{([\gamma+\delta]-2)!} \qquad (3)$$

permutations. But only **n** curves are used for mobile grid computing.

4. User 1 and BS define the HC binary tree (HCBtree) according to look-up table 1 in the following form: a node of HCBtree is composed by three data components (hyperelliptic curve index τ , f_3 and f_0) and two pointers to left child and right child.

5. BS assigns a secret HC C_τ to User τ and BS constructs a look-up table 2, using the grid resources, for each secret HC C_τ. Where $\boldsymbol{K}_\tau = \#\boldsymbol{J}(C_\tau)$ is the Jacobian cardinality $\boldsymbol{J}(C_\tau)$ over $GF(2^n)$.

6. For($\tau = 2;\ \tau \le \boldsymbol{n}\ ;\ \tau + 1$) Go through the HCBtree in post-order:

User 1 and User τ use the Diffie-Hellman scheme with the HC C_τ.

1. User 1 selects the reduced divisor $Q = D_{\tau l}$, which have an identifier number Id^l according to look-up table 2 of HC C_τ for User τ.
2. User 1 sends the identifier number Id^l to User τ. And this number is represented by binary sequence $c_{\tau l}$. Then, sequence $c_{\tau l}$ is encoded using a LDPC code and the codeword $U_{\tau l}(u)$ is obtained.
3. $U_{\tau l}(u)$ is modulated with M-PSK modulation and $X_{\tau l}$ is obtained.
4. $X_{\tau l}$ is corrupted in the fading channel and $W_{\tau l}$ is generated.
5. $W_{\tau l}$ is demodulated with M-PSK demodulation and $\hat{U}_{\tau l}(u)$ is obtained.
6. $\hat{U}_{\tau l}(u)$ is decoded using a LDPC code and $\hat{c}_{\tau l}$ is generated.
7. User τ obtains the identifier number Id^l from $\hat{c}_{\tau l}$. Then, User τ obtains the reduced divisor $Q = D_{\tau l}$.
8. User 1 selects a random integer secret key $k_{1\tau}$ and calculates its public key $k_{1\tau}Q$.
9. User 1 repeats steps 2 to 7 and User τ obtains $k_{1\tau}Q$.
10. User τ selects a random integer secret key $k_{\tau\tau}$ and calculates its public key $k_{\tau\tau}Q$.
11. User τ repeats steps 2 to 7 and User 1τ obtains $k_{\tau\tau}Q$.
12. User 1 calculates $P = k_{t\tau}(k_{r\tau}Q) = div\left(a^{ck,\tau}(u), b^{ck,\tau}(u)\right)$ and User τ calculates $P = k_{r\tau}(k_{t\tau}Q) = div\left(a^{ck,\tau}(u), b^{ck,\tau}(u)\right)$.

End_For.

3. Results

The system security is based on many discrete logarithm problems. In the system is used the Diffie-Hellman scheme and the common secret key $k_r(k_t D_l)$ is calculated by an efficient exponentiation and its secret is guaranteed by the intractability of the discrete logarithm problem. Then, in the algorithm 1, the HC family $\{C_\tau\}_{\tau=1}^{Per}$ is hidden because $k_r(k_t D_l)$ is secret.

In the algorithm 1 the dominant steps are computing order of Jacobian, LDPC decoding, reduced divisor addition and reducing divisor and searching in a look-up table. Then, $\boldsymbol{K} = \#J(C)$ is calculated in polynomial complexity $O\left(2^{4+\varepsilon}\boldsymbol{n}^{3+\varepsilon}\right)$, where the HC C is defined over $GF(2^n)$and $\varepsilon > 0$ is a small real number [8]. The LDPC decoding algorithm has polynomial complexity, which is $O(N)$ over $GF(2)$ [8].The identifier number Id^l of $D_l = div(a_l(u), b_l(u))$ is obtained from a look-up table and this searching has complexity $O(Per)$ or $O(\boldsymbol{K})$. A reduced divisors addition has complexity $O(\vartheta(\boldsymbol{g})\log \boldsymbol{g})$ and a divisor reduction has complexity $O(\boldsymbol{g}\,\vartheta(\boldsymbol{g}))$, where $\vartheta(\boldsymbol{g})$ is the number of operations in $GF(2^n)$ for multiplying two polynomials of degree at most **g** [8], **g** is the genus of C. Therefore,

the complexity of algorithm 1 is polynomial and given by $O(Per*\vartheta(\boldsymbol{g}))$ where genus $\boldsymbol{g} = 2$.

Figure 2 shows the comparison curves of BER v/s SNR for proposed system against the LDPC code over fading channel with BPSK modulation and against the Fading channel with BPSK modulation. This system produces an improvement of 0.82 dB at a BER = 10^{-4} can be achieved when it is compared with the LDPC code over fading channel with BPSK modulation.

4. Conclusions

The proposed system allows key exchange and reliable data transmission through the mobile grid computing environment. The system uses very strong one way functions based on the combination of a family of HC and a LDPC code.

The security of the whole scheme is guaranteed by the intractability of the discrete logarithm problem on the generator curve C and the curves C_τ.

The implementation of the algorithm based on the concatenation of a family of HC and a LDPC code has a polynomial time complexity expressed by $O(Per*\vartheta(\boldsymbol{g}))$, where *Per* is the number of secret HC.

Finally, the proposed system produces an improvement of 0.82 dB at a BER = 10^{-4} which is achieved when it is compared with the LDPC code over Rayleigh Fading channel with BPSK modulation.

Acknowledgements

The authors would like to thank PBCT CONICYT ACT11/04 -Chile, for their financial support.

References

[1] Foster, I.; Kesselman, C.; Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International J. Supercomputer Applications, 15(3), 2001.

[2] Foster, I.; Kesselman, C. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kauffman, 2003.

[3] Minoli, D. A Networking Approach to Grid Computing. ISBN 0-471-68756-1, John Wiley & Sons, Inc. 2005.

[4] Ramakrishnan, L. Securing Next-Generation Grids. IT PRO, IEEE Computer Society, March/April 2004 pp. 34-39.

[5] Schneier, B., Applied Cryptography. ISBN 0-471-11709-9, Second Edition, John Wiley & Sons, Inc.1996.

[6] Koblitz, N., Hyperelliptic Cryptosystems, Journal of Cryptology, N º 1, 1989, pp 139 – 150.

[7]Koblitz, N., Algebraic Aspect of Cryptography. Algorithms and Computation in Mathematics. ISBN 3-540-63446-0 Springer-Verlag, 1998.

[8] Jacobson M., Jr., Menezes A., Stein, A., Hyperelliptic Curves and Cryptography. Available at: http://pages.cpsc.ucalgary.ca/~jacobs/publications.html

[9] Lange, T., Efficient arithmetic on genus 2 hyperelliptic curves over finite fields via explicit formulae, 2003. Available at http://www.itsc.ruhr-uni-bochum.de/tanja

[10] Gallager, R. G, Low-density parity-check code, IRE Trans. Inform. Theory, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

[11] Sklar, B, "Digital Communications Fundamentals and Applications". ISBN 0130847887. Second edition, 2001. Prentice Hall, Inc.

[12] Davey, M. C., MacKay, D. J. C, Low Density Parity Check Codes over GF(q), Proceedings of the IEEE Information Theory Workshop, 1998, Killarney.

[13] Fulton, W. "Algebraic curves. An Introduction to Algebraic Geometry". W.A. Benjamin, Inc, N.Y. 1969.

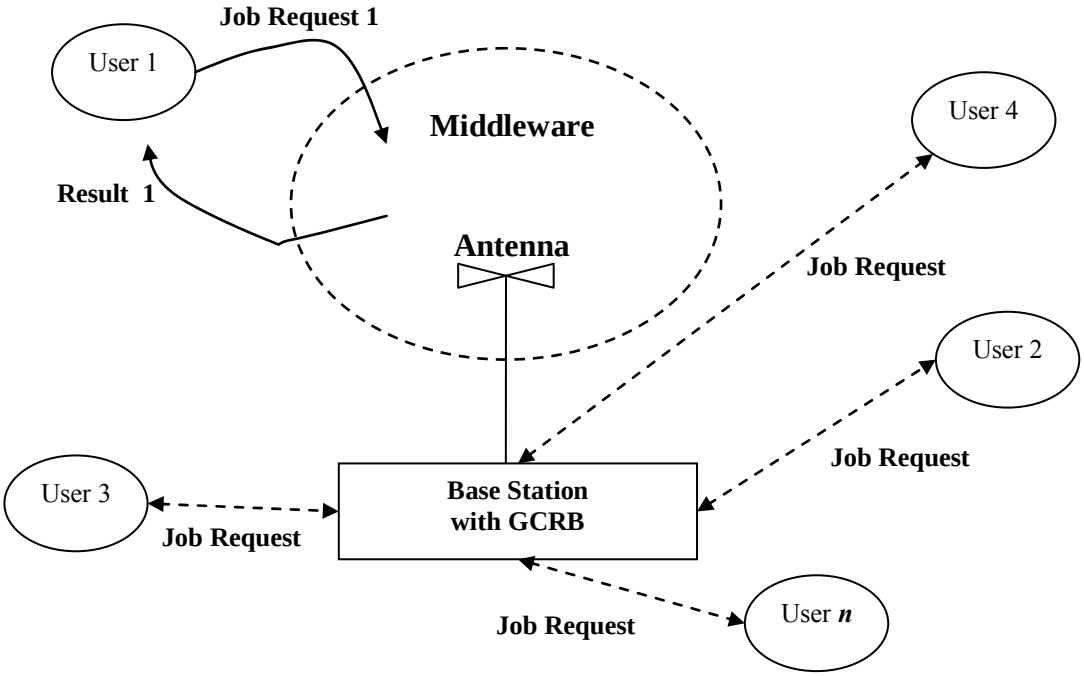

Figure 1. Mobile Grid Computing environment.

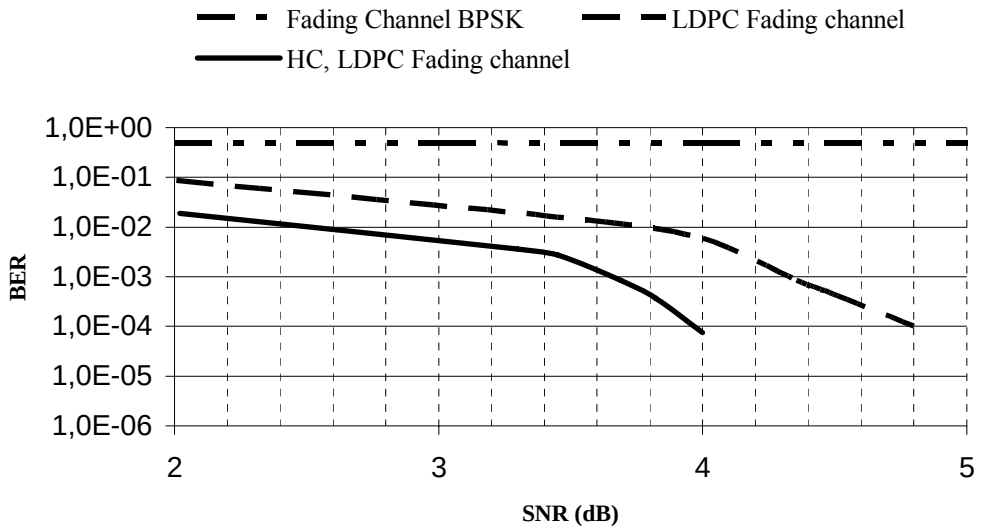

Figure 2. Comparison curves of BER v/s SNR for proposed system against LDPC over fading channel and Fading channel BPSK.

HC index τ	f_3	f_0
1	ω_{l_i}	ω_{l_j}
…	**…**	**…**
Per	ω_{l_p}	ω_{l_o}

Look-up table 1. Coefficients f_3 and f_0

Identifier number	**Reduced divisor**
1	$D_{\tau 1} = div\left(a_{\tau 1}(u), b_{\tau 1}(u)\right)$
…	…
$\boldsymbol{K}_\tau$	$D_{\tau \boldsymbol{K}_\tau} = div\left(a_{\tau \boldsymbol{K}_\tau}(u), b_{\tau \boldsymbol{K}_\tau}(u)\right)$

Look-up table 2. Identifier numbers and reduced divisors for HC C_τ.