International Journal of Hybrid Information Technology
Vol.7, No.1 (2014), pp.137-146
http://dx.doi.org/10.14257/ijhit.2014.7.1.11

Comparative Study of Hyperelliptic Curve Cryptosystem over Prime Field and Its Survey

P. Vijayakumar[1], V. Vijayalakshmi[2] and G. Zayaraz[3]

[1]*Research Scholar, Department of ECE, Pondicherry Engineering College*
[2]*Assistant Professor, Department of ECE, Pondicherry Engineering College*
[3]*Associate Professor, Department of CSE, Pondicherry Engineering College*

[1]*vijayrgcet@gmail.com;* [2]*vvijizai@pec.edu;* [3]*gzayaraz@pec.edu*

Abstract

Public key cryptography is the famous cryptography technique used in many corporate sectors for developing software to provide security services. Hyperelliptic Curve Cryptosystem (HECC) is one of the public key cryptographic technique, an expansion of Elliptic Curve Cryptography which offers the similar level of security compared with other cryptosystems such as RSA, ECC and DSA. HECC supervise the ECC due to shorter operand size. This paper will analysis the performance of HECC over genus curve 2, 3, 4, 5, 6 and compares the result with ECC. It also inspects the use of Hyperelliptic Curve Cryptography technique and affords complete survey on its performance for many applications.

Keywords: *Public Key Cryptography, Elliptic Curve Cryptography, Hyperelliptic Curve Cryptography, RSA, Discrete logarithm Problem, Genus curve, processing time*

1. Introduction

Cryptography is the art of science which is used to encrypt and decrypt the data for secure communication mathematically. It provide facility to the user to transfer data securely without degrade the performance of the system. Public key cryptography is one of the cryptographic techniques which consist of pair of keys known as private and public key. Public key is used to encrypt the data and private is used to decrypt the data. Hyperelliptic curve cryptography is the fast public key cryptographic technique with high efficiency and security [1, 2]. In 1988, Neal Koblitz suggested a new higher genus curve for cryptographic purpose known as Hyperelliptic Curve Cryptosystem. HECC has more advantage such as shorter key size, less computational overhead, high security, require less memory space and consume less power. These features makes easy to implement HECC both in hardware and software. Since HECC has enormous feature for providing security and high efficiency for engineering application. Many researchers put his effort to develop cryptographic algorithm and protocol based on Hyperelliptic Curve Cryptosystem. This feature makes HECC very popular among the many cryptographic system [3-5].

This paper is organized as follows: Section 2 gives a brief overview of the mathematical background of Hyperelliptic Curve Cryptography. Section 3 introduces the implementation of Hyperelliptic Curve cryptosystem over prime field of genus curve 2, 3, 4, 5 and 6. Section 4 summarizes the simulation results and provides the comparison table. Finally, end this paper with discussion of result and conclusion.

ISSN: 1738-9968 IJHIT
Copyright ⓒ 2014 SERSC

2. Mathematical background of Hyperelliptic Curve Cryptography

The security of Hyperelliptic Curve Cryptosystem depends on the discrete logarithm problem. This problem helps to avoid the eavesdropper from breaking of keys even both Q and P values are known publicly. Different types of curve have to study to understand about public key (Q), group point (P) and Hyperelliptic Curve Discrete Logarithmic problem (HECDLP) [6, 7].

2.1. Hyperelliptic curve

Hyperelliptic curve [3] E of genus g>=1 over finite filed F is the set of solution (x,y) ϵ F x F to the equation

$$E: y^2 + h(x)y = f(x); \tag{1}$$

where h(x) is a polynomial of degree g and h(x) ϵ F(x).
f(x) is a monic polynomial of degree 2g+1 and h(x) ϵ F(x).

The curve E is said to be non-singular curve, if there are no pairs (x ,y) ϵ F x F. The polynomial f(x) and h(x) are chosen such that it has to satisfy the following equations

$$2y + h(x) = 0 \tag{2}$$
$$h'(x)y - f'(x) = 0 \tag{3}$$

2.2. Types of genus curve

Genus curve decide the processing time of the Hyperelliptic Curve Cryptosystem such as key generation, encryption and decryption process. Value of g decided the polynomial of curve E like g = 2, 3, 4. Polynomial chosen for genus 2, 3, 4, 5 and 6 over prime field F_p is given below

Genus g=2
$$y^2 = x^5 + a_3x^3 + a_2x^2 + a_1x + a_0 \tag{4}$$
Genus g=3
$$y^2 = x^7 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 \tag{5}$$
Genus g=4
$$y^2 = x^9 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 \tag{6}$$
Genus g=5
$$y^2 = x^{11} + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 \tag{7}$$
Genus g=6
$$y^2 = x^{11} + a_9x^9 + a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 \tag{8}$$

2.3. Jacobian of Hyperelliptic Curve

The Jacobian of curve E defined over finite field F is denoted by $J_E(F)$. Each elements of the Jacobian can be represented uniquely by a divisor D as shown Eq.(4) and Eq.(5).

$$J_E(F) = \frac{D^0}{P} \tag{9}$$
$$D = \sum m_i P_i \tag{10}$$

 Copyright ⓒ 2014 SERSC

Where D is called as reduced divisor; m_i – Number of points ; P_i – Points on the Curve E
The reduced divisor D is represented using Mumford representation which forms the group law[8].

2.4. Mumford representation

Mumford [9] representation that the divisors of the Jacobian can be represented as pair of polynomials u(x) and v(x) which satisfies the following condition:

- u(x) is monic polynomial;
- deg v(x) < deg u(x) < = g;
- (u(x))/(y^2+h(x)y-f(x)) = 0

2.5. Cantor's algorithm

The cantor algorithm is the basic algorithm for performing arithmetic group operation such as divisor adding and doubling in the Jacobian of Hyperelliptic Curve [8]. These group operations will be performed in two steps. In first step, find the reduced divisor D'=div(u',v') such that D'=D1+D2 = div(u1,v1)+div(u2,v2) in the group $J_E(F)$.In second step, find the semi reduced divisor D'=div(u',v') to an equivalent divisor D=(u , v). Algorithm 1 and 2 explains about group addition and divisor doubling operation respectively as shown below:

2.5.1 Algorithm 1: Group Addition operation add two divisor D1 and D2, results are store in divisor D. The following steps have to follow to obtain divisor D.

Input : $D_1 = div(u_1, v_1), D_2 = div(u_2, v_2)$
Output : D = $div(u_3, v_3) = D_1 + D_2$
Step 1 : $d = GCD(u_1, u_2, v_1 + v_2 + h)$
$= s_1u_1 + s_2u_2 + s_3(v_1 + v_2 + h)$
Step 2: $u' = u_1u_2d^{-2}$
Step 3: $[s_1u_1v_2 + s_2u_2v_1 + s_3(v_1v_2 + f)]d^{-1}(mod\, u')$
Step 4 : $k = 0$
Step 5: $while \deg u'_k > g \;\; do$
Step 6 : k= $k + 1$
Step 7 : $u'_k = \frac{f - v'_{k-1}h - (v'_{k-1})^2}{u'_{k-1}}$
Step 8:v'_k=$(-h - v'_{k-1})\ mod\ u'_k$
Step 9:end while
Step 10: output $(u_3 = u'_k, v_3 = v'_k)$

2.5.2 Algorithm 2: Doubling a divisor is easier than general addition and algorithm 1 can be simplified as shown below:

Step 1: $d = \gcd(u, 2v + h) = s_1u + s_3(2v + h)$
Step 2: $u' = u^2d^{-2}$
Step 3: $v' = [s_1uv + s_3(v^2 + f)]d^{-1}(mod\, u')$

Copyright ⓒ 2013 SERSC

3. Hyperelliptic Curve Cryptosystem

The divisor doubling and addition operation are used to implement Hyperelliptic Curve Cryptosystem. HECC consist of three processes such as key generation, encryption and decryption. These processes involved in divisor generation as shown in Equation (10) by choosing proper polynomial of genus curve 2, 3, 4, 5 and 6 as shown in Equation (4), (5), (6), (7) and (8). ElGamal method is used to design Hyperelliptic Curve Cryptosystem process, known as HEC-ElGamal algorithm as shown below:

3.1. Algorithm for public & private key generation

Input: The public parameters are Hyperelliptic curve C, prime p and divisor D.
Output: The public key P_A and private key a_A.
Process:

- *Private key*: $k_A \in RN$; Random prime number k_A is chosen in order of N.
- *Public key*: $P_A \leftarrow k_A . D$;
  P_A *is represented as pair of polynomial* $[(u(x), v(x))]$ and D is Divisor
- *Key pair* : $[(k_A, P_A)]$

3.2. Encryption algorithm

The plaintext 'm' is converted into ASCII value and these values are represented as sequence of points (u_x, v_y). The encoded message is referred as E_m. The following steps are followed to encrypt the encoded message E_m of user A and send it to user B.

- *Private Key*: $k_A \in RN$; Random prime number k_A is chosen in order of N.
- *Public key*: $P_A \leftarrow k_A . D$;
  P_A *is represented as pair of polynomial* $[(u(x), v(x))]$ and D is Divisor
- *Agreed key*: $Q_A \leftarrow k_A . P_B$; P_B is represented as receiver's public key.
- *Cipher text* :$C_m \leftarrow \{Q_A, E_m + P_A\}$; C_m *is represented as* $[(u(x), v(x))]$.

3.3. Decryption algorithm

To decrypt the cipher text C_m, user B extracts the first coordinate 'Q_A' from the cipher text then multiply with its private key (a_B) and subtract the result from the second coordinate. This can be written as follows:

$$
\begin{aligned}
&E_m + \mathrm{k}\, P_B - a_B(Q_A) = E \\
&= E_m + kP_B - k(a_B D) \\
&= m + kP_B - a_B(kD) \\
&= E_m + kP_B - kP_B = E_m
\end{aligned} \tag{11}
$$

User 'A' masked the message E_m by adding $k.P_B$ to it. But user 'A' only knows the value of k, so even though P_B is a public key, nobody can remove the mask $k.P_B$. For an attacker to remove message, the attacker would have to compute k from the given D and [k] D, *i.e.*, Q, which is assumed to be very hard. This process remembers the concept of Hyperelliptic Curve Discrete Logarithmic Problem (HECDLP)[10].

 Copyright ⓒ 2014 SERSC

4. Performance Analysis

The Hyperelliptic Curve Cryptosystem for genus curve 2,3,4,5 and 6 was implemented in MATLAB R2012b version and executed in Intel(R) Core(TM) i5-3230M CPU @2.60GHz with 4GB internal RAM. The parameters taken to test the system are processing time and key size. Simulation result shows that the time taken to generate divisor, pair of key, encryption and decryption of data. The equations Eq(4), Eq(5), Eq(6), Eq(7) and Eq(8) are implemented in MATLAB to generate divisor, key generation, encryption and decryption. Table 1 and Table 2 values are obtained by checking the processing time of each every function by using simulation tool have option "Run and Time" which shows simulation time to process the functions involved in algorithm.

Table 1. Comparison of genus g value for length of prime = 45

Process	Processing Time(ms)				
	g=2	g=3	g=4	g=5	g=6
Divisor generation	710	918	1171	1221	1871
Key Generation	2343	3556	8026	8860	11745
Encryption	3305	5456	8322	8622	9322
Decryption	3505	6534	10595	10700	11595

Table 2. Comparison of genus g value for length of prime = 65

Process	Processing Time(ms)				
	g=2	g=3	g=4	g=5	g=6
Divisor generation	871	1453	1771	2013	2313
Key Generation	6860	7867	8860	11743	15743
Encryption	3305	6466	8322	8997	12109
Decryption	3505	7867	10595	11100	15012

Figure 1 and Figure 2 shows the comparative graph for polynomial over genus 2, 3, 4, 5 and 6 values for length of prime 45 and 65 respectively. From these graph, it is inferred from that result genus g = 2 be the best curve for Hyperelliptic Curve Cryptosystem compared with other genus g values.

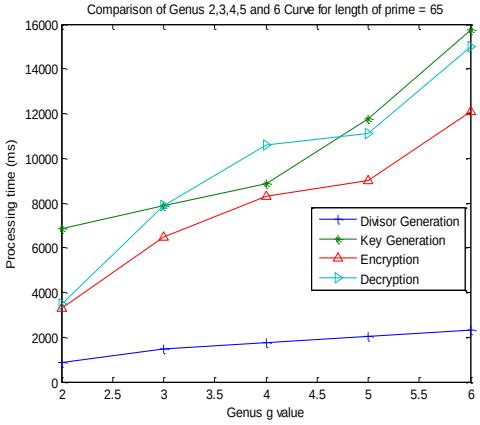

Figure 1. Genus g Vs processing time for length of prime = 65

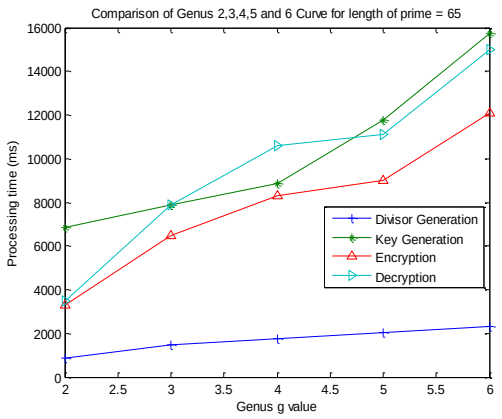

Figure 2. Genus g Vs processing time for length of prime = 45

Copyright ⓒ 2013 SERSC

5. HECC Survey and its Application

Avanzi M [11] has proved that HECC over prime field is satisfactory enough when compared with Elliptic Curve, especially when large point groups are desired. Fan and Gong [12] also proved that HECC provides greater efficiency than either integer factorization systems or discrete logarithm systems, in terms of computational overheads, key sizes, and bandwidth. Deng Jian-zhi [13] illustrate that design of HECC based Digital signature which will solve the problem by checking the integrity of the file and signature ID, and it especially fit for the internet operation which need identity validate. It also grants the hardware module for HEC-DSA signature and validates using the simulator Quartus II 6.0. S.Baktir [14] launches the implementation of HECC over an extension field of odd characteristic on an embedded processor and proves that iimplementation of genus-2 HECC over GF(2^{81}) on 32-bit ARM7TDMI processor provides an improvement of approximately 57% compared with other cryptographic algorithm. It also furnishes the software implementation of HECC over OTF using Microsoft's Visual C++ Compiler 6.0 and Developer Studio 6 for writing the program for Compilation and debugging the field multiplication, field inversion and group addition and doubling. Prasanna Ganesan highlights the usage of Hyperelliptic Curve Cryptography for power constrained mobile devices and measure the performance of both RSA algorithm and HECC using PALMIII and J2ME wireless tool kit 2.5.1 respectively. It also advice to achieve the high security level by using ElGamal based Hyper Elliptic curve cryptography and MD5 algorithm in authentication protocol [15].

Michael Jacobson [16] observes that Jacobian of a Hyperelliptic curve defined over a finite field implement the discrete logarithm cryptographic protocols. It suggested a Weil Descent technique to solve Hyperelliptic Curve Discrete Logarithmic Problem (HCDLP). It also gives detail about index calculus attacks on HCDLP and Weil descent attack on the ECDLP. Alexander Klimm developed a Crypto processor to authenticate the automotive access control systems to unlock the car. It also design the hardware and software interface for automotive access control system [17]. Kakali Chatterjee explores the details of main operations like scalar multiplication, group operations on Jacobian, finite field operations etc for efficient implementation of ECC / HECC. It also compares the timing of operations like scalar multiplication, encryption and decryption of ECC and HECC. It examined the performance of HECC on PC with Intel Core 2DUO CPU T6400@2.00GHz with 4GB RAM and windows vista operating system using jdk1.6. Software implementation of ECC and HECC for constrained devices such as smart card, PDA, and mobile are done. Finally it gives the performance comparison of ECC and HECC [18]. Wollinger analysis the Explicit Formulae for Hyperelliptic Curve Cryptosystems which perform the group addition and doubling operation of HECC based on Cantor or Harley algorithm [19]. Xuanwu Zhou proves that Ring signature effectively solves the problem of group managing in group cryptosystem and thus greatly improves the efficiency of signature generating and verifying. It also shows the improvement in ring signature then reinforces the security and stability of ring signature, and effectively improves the efficiency of engineering application [20].

5.1. Security of HECC

Neal Koblitz first shows the Hyper-Elliptic Curve cryptosystem as the expansion of ECC. Hyper-Elliptic curve cryptosystem is the natural generalization of Elliptic curve cryptosystem. And the security of HCC is based on HCDLP. For any $k \in Z_l^*$, the computation of $K = k.P$ via k and P is computationally feasible; but the computation of k via K and P is HCDLP, it is computationally infeasible. Hyper Elliptic Curve cryptosystem has been followed by superiorities in security and efficiency.

 Copyright ⓒ 2014 SERSC

- Cryptosystem based on Hyper-Elliptic Curve Jacobian group has the same security level as cryptosystem based on Elliptic Curve rational point group with the same group order.
- Hyper-Elliptic Curve cryptosystem can acquire the same security level with shorter operating parameters. If the basic finite field is 60 bits, the security level of Hyper-Elliptic Curve cryptosystem is equivalent to ECC with 180 bits, and it secure more than RSA with 1024 bits.
- At present, the attack algorithms against Hyper-Elliptic Curve cryptosystem with low genus ($4 \leq g$) all prove to be inapplicable with exponent complexity.
- In Hyper-Elliptic Curve cryptosystem, secure Jacobian group with large prime number order can be constructed on a relatively smaller basic field.
- By the same domain field, the bigger genus ($4 \leq g$), is more the number of curves, therefore it is much easier to choose secure Hyper-Elliptic Curve to construct a cryptosystem.

5.2. Software and hardware implementation of HECC

5.2.1 8051 microprocessor: HECC is implemented on 8051 processor by two types. The first type is a pure software implementation - either a pure C model operating on 8051 or mixed with C/assembly model in most of the functions are performed in C, while GF(2^{83}) finite field multiplier is performed in assembly. The second type is mixed with hardware/software models in which some of the functions are performed in C while the GF (2^{83}) finite field operations (multiplication/addition/inversion) are performed in hardware. The hardware operators and 8051 are connected by a memory-mapped interface, over the 8051's P0, P1, and P2 I/O port interfaces [24].

5.2.2. GEZEL: GEZEL is a cycle-based hardware description language based on Finite-State-Machine + Data path (FSMD) model. The GEZEL tools offer stand-alone simulation, cosimulation, and code-generation into synthesizable (VHDL) code. Through user-defined library-block extensions in C++, new cosimulation interfaces can be added. GEZEL is open-source [25].

5.2.3. Quartus II 6.0: Quartus II 6.0 used to implement the algorithm based on Hyperelliptic Curve cryptography. It can able to generate the RTL circuit and simulated waveform which is useful for checking the circuit connection in chip. RTL is a top-to-down structure. It is a open source software.

5.3 Discussion and recommendation

HECC offer higher level of security with less computation power and less processing time compared with ECC and RSA technique. It will be implemented using 32 bit microprocessor and applicable for many power constrained devices such as smart card, sensors, mobiles, PDA etc,. It affords greater level of security in many wireless system especially military applications, Weather monitoring, banking system and E-Commerce. HECC having many choice of genus curve such as genus 2, 3, and 4 which will give hierarchical level of security. Among these genus curve, genus 2 provide the enhanced level of security with less power computation overhead and less communication overhead.

Copyright ⓒ 2013 SERSC

6. Conclusion

This paper shows the performance of HECC over genus curve 2, 3, 4, 5, and 6 for length of prime field F_q = 45 and 65. Since HECC has the advantage of shorter operand length than ECC and offers same security level. This is often used for power constrained devices to improve the secrecy of data and efficiency of the system. Jacobians of Hyperelliptic curves of genus 2 or 3 require less processing time for divisor generation, key generation, encryption and decryption than genus 4, 5 and 6. Finally, this paper summarizes the survey on HECC application, security requirement and explains about simulation tool.

References

[1] W. Diffie and M. Hellman, "New directions in cryptography", in the IEEE Transaction on Information Theory, vol. 22, issue 6, **(1976)** November, pp. 472 - 492.
[2] W. Stallings, "Cryptography and Network Security Principals and Practices", Pearson edition (India) Pvt.ltd, 4th Edition, **(2009)**.
[3] N. Koblitz, "A Family of Jacobians Suitable for Discrete Log Cryptosystems", Advances in Cryptology Crypto '88, **(1988)**, pp. 94-99.
[4] Y. Sakai and K. Sakurai, "Design of Hyperelliptic Cryptosystems in Small Characteristic and a Software Implementation over IF^{2n}", Advances in Cryptology Proc. ASIACRYPT '98, **(1998)**, pp. 80-94.
[5] J. Pelzl, "Hyperelliptic Cryptosystems on Embedded Microprocessor", master's thesis, Dept. of Electrical Eng. and Information Sciences, Ruhr-Universitaet Bochum, Bochum, Germany, **(2002)** September.
[6] N. Koblitz, "Hyperelliptic Cryptosystems", in the Journal of Cryptology, vol. 1, no. 3, **(1989)**, pp. 129-150.
[7] N. Koblitz, "Algebraic Aspects of Cryptography", in the Springer series of Algorithms and Computation in Mathematics, vol. 3, **(1998)**.
[8] D. Cantor, "Computing in Jacobian of a Hyperelliptic Curve", in the journal of Mathematical Computation, vol. 48, no. 177, **(1987)** January, pp. 95-101.
[9] D. Mumford, "Tata Lectures on Theta II", in the Springer journal of Progress in Mathematics, vol. 43, **(1984)**.
[10] N. Smart, "On the Performance of Hyperelliptic Cryptosystems", in the Springer LNCS proceedings of Advances in Cryptology- EUROCRYPT '99, vol. 1592, **(1999)**, pp. 165-175.
[11] M. Avanzi, "Aspects of Hyper-Elliptic Curve over large prime fields in software implementations", in the Springer LNCS proceedings of Cryptographic Hardware and Embedded Systems, vol. 3156, **(2004)**, pp. 148-162.
[12] X. Fan, and G. Gong, "Efficient Explicit Formulae for Genus 2 Hyper Elliptic Curve over Prime Fields and Their Implementations", in the Springer LNCS proceedings of Selected Areas in Cryptography, vol. 4876, **(2007)**, pp. 155-172.
[13] D. Jian-zhi, C. Xiao-hui and G. Qiong, "Design of Hyper Elliptic Curve Digital Signature", in the IEEE proceeding of International Conference on Information Technology and Computer Science, vol. 2, **(2009)** July, pp. 45-47.
[14] S. Baktir, J. Pelzl, T. Wollinger, B. Sunar and C. Paar, "Optimal Tower Fields for Hyperelliptic Curve Cryptosystems", in the IEEE transaction of Signals, Systems and Computers, vol. 1, **(2004)**, pp. 522-526.
[15] S. P. Ganesan, "An Authentication Protocol For Mobile Devices using Hyper Elliptic Curve Cryptography", in the ACEEE proceeding of International Journal of Recent Trends in Engineering and Technology, vol. 3, no. 2, **(2010)** May.
[16] M. Jacobson Jr., A. Menezes and A. Stein, "Hyper Elliptic Curve and Cryptography", in the proceedings of Fields Institute Communications, vol. 41, **(2004)**, pp. 1-27.
[17] A. Klimm, "A Flexible Integrated Crypto processor for Authentication Protocols based on Hyperelliptic Curve Cryptography", in the IEEE proceeding of International Symposium on System on Chip (SoC), vol. 1, **(2010)**, pp. 35-42.
[18] K. Chatterjee, A. De and D. Gupta, "Software Implementation of Curve based Cryptography for Constrained Devices", in the proceedings of International Journal of Computer Applications (0975 – 8887), vol. 24, no. 5, **(2011)** June.
[19] T. Wollinger, J. Pelzl and C. Paar, "Cantor versus Harley: Optimization and Analysis of Explicit Formulae for Hyper Elliptic Curve Cryptosystems", in the IEEE transaction of Computers, vol. 54, issue 7, (2004) July, pp. 861- 872.

 Copyright ⓒ 2014 SERSC

[20] X. Zhou, Xi'an and Tianjin , "Improved Ring Signature Scheme Based on Hyper-Elliptic Curve", *in* the IEEE proceeding of Second International Conference on Future Information Technology and Management Engineering, (2009) December, pp. 373-376.

Authors

P. VIJAYAKUMAR is currently working as Assistant Professor (Sr.) in School Electronic Science Engineering at VIT university Chennai campus, India and pursuing Ph.D in Pondicherry University. He completed his B.Tech in Rajiv Gandhi College of Engineering and Technology and M.Tech in Pondicherry Engineering College which is affiliated to Pondicherry University. He has 7 years of teaching experience. To his credit, he has published more than 15 research papers relating to Cryptography and Network Security in several National / International Journals and Conferences. He can be reached by email at vijayrgcet@gmail.com

Dr. V. Vijayalakshmi is currently working as Assistant Professor in Electronics & Communication Engineering Department at Pondicherry Engineering College, Puducherry, India. She completed her B.Tech, M.Tech and PhD in Pondicherry Engineering College which is affiliated to Pondicherry University. She has 20 years of teaching experience. To her credit, she has published more than 25 research papers relating to Network Security and software Engineering in several National / International Journals and Conferences. She can be reached by email at vijizai@pec.edu

Dr. G. Zayaraz is currently working as Professor in Computer Science & Engineering Department at Pondicherry Engineering College, Puducherry, India. He received his Bachelor`s, Master's and Doctorate degree in Computer Science & Engineering from Pondicherry University. He has published more than twenty five research papers in reputed International Journals and Conferences. His areas of specialization include Software Architecture and Information Security. He is a reviewer for several reputed International Journals and Conferences and Life Member of CSE, and ISTE. He can be reached by email at gzayaraz@pec.edu

Copyright ⓒ 2013 SERSC

Copyright ⓒ 2014 SERSC