Secure Mobile Ad Hoc Routing Using Confrontations (SMARUC) and Nodes Communication with CCM (Character Classification Model) - OKE (Optimal Key Exchange) - SSL (Secure Socket Layer) Model

R. M. Krishna Sureddi[1(✉)], Santosh Kumar Ravva[2], and Ramakrishna Kolikipogu[3]

[1] Department of IT, Chaitanya Bharathi Institute of Technology, Hyderabad, India
rmkrishnsureddi_it@cbit.ac.in
[2] Department of CSE, Vasavi College of Engineering, Hyderabad, India
santosh@staff.vce.ac.in
[3] Department of IT, Chaitanya Bharathi Institute of Technology(A), Hyderabad, India
ramakrishna_it@cbit.ac.in

Abstract. Manet (Mobile Adhoc network) is a dynamic network with no fixed infrastructure. So Data Routing among nodes in a wireless network is pretty complex when compared to traditional centralized network architecture. There is always possibility the nodes vulnerable to all kinds of attacks. Even though cryptography models, key exchange models are there for data exchange but all are struggled with security and all are bound by global time complexities even in message communication among nodes. So we addressed this problem by generating trusted dynamic topology with secure data communication using "SMARUC with CCMOKE" Model. This method which generates exchanges key in minimum time complexity even compared to Diffie-Hellman model. For Crypto, data transmission between nodes done with simple RSC-OKE-SSL crypto model which performs communication in minimum time complexity compared to other cryptography models. By this method of forming dynamic network blocks the doorway of malicious nodes and generates the secured dynamic topology.

Keywords: Cryptography · Keyexchange · SSL · classification · Roughset · encryption · decryption · Hybrid crypto models · entropy · randomness · floating frequency · Crypto tool

1 Introduction

A mobile ad hoc network (MANET) [1] is self configuration autonomous system of mobile nodes. Mobile Nodes are connected by wireless links, the union of which forms a network in the form of a dynamic communication structure. The routers are permitted to move at any speed in any direction and establish themselves indiscriminately. Thus, the network's wireless topology may dynamically vary in an unpredictable manner. There is no fixed arrangement, and information is forwarded in peer-to-peer mode using multi hop routing. MANETs [1–8] are basically peer-to-peer (p2p) multi hop mobile wireless

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023 Workshops, LNCS 14105, pp. 160–173, 2023.
https://doi.org/10.1007/978-3-031-37108-0_11

networks where information packets are transferred in a store-and-forward method from source to destination, via intermediate nodes, as shown in Fig. 1. As the nodes move, the resulting change in network topology must be made known to the other nodes so that prior topology information can be updated. Such a network may operate in a stand-alone fashion, or with just a few selected routers communicating with an infrastructure network.

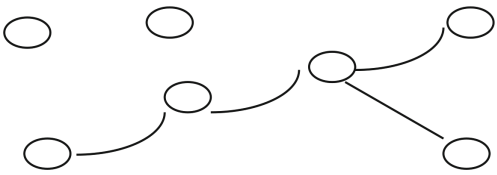


Fig. 1. Illustration of a MANET

a. **Characteristics of MANETs**

- No integrated control and administration is done
- Communication through the various nodes in a network
- Alteration of topology and frequent breakage of links once the connection is proven
- Can be set up anywhere in any situation
- Limited security

b. **Security issues in MANETs**

1) **Easier to hit:** Since the broadcasting is nothing but air, it can be tapped easily.
2) **Quality of Service:** Quality of Service (QoS) guarantee is very much essential for the successful communication of nodes in the network. The different QoS metrics includes throughput, packet loss, delay, and jitter and error rate. The dynamically changing topology, limited bandwidth and quality makes the difficulty in achieving the desired QoS guarantee for the network.
3) **Limited Security:** Easily influenced by all crypto attacks.

1.1 Objective

The main objectives of this paper are

- To generate dynamic secured topology with SMARUC [1,9] Model
- To generate dynamic trusted network in optimized key exchange Model.
- Establish communication among nodes in simple with efficient crypto model.
- To provide secure communication among trusted Nodes through double encryption model.
- Protecting the network from malicious attacks.

1.2 SMARUC^2 Algorithm Description

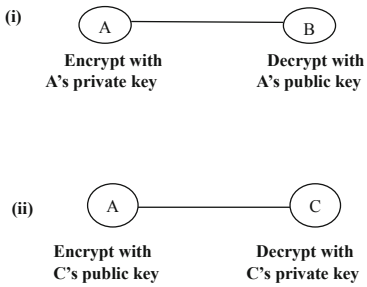
Secure Mobile Ad Hoc Routing [4] Confrontations (SMARUC) [2] accomplishes establishment of companion networks in MANETs in the same way as in real life scenarios.

1) Confrontation Your Neighbor

Confrontation is a method to authorize nodes primarily when no stipulation is present. It is a basic test which a node has to complete in order to corroborate its honesty and integrity. Let us assume that the node A confronts its neighbor node B.

Step 1) When a new system is initialized, every node is a foreigner to a different. Thus each node sends its neighbors in the unconfirmed list.
Step 2) The node A picks one of the neighbors, B and performs the usual Share associates Stage.
Step 3) As a reply the neighbor node B either sends its companion directory or the nodes from its unsubstantiated directory if the companion list is empty.
Step 4) On receiving the list, the node A picks up a node which can reach on its own. Let us say that this node is C.
Step 5) Now the node A has two ways to reach the node one through B and another through a route already known to it.
Step 6) The above all 6 steps achieved through - Each node is initialized with a pair of large prime numbers.
Step 7) Node A wants to confront node B. Then it sends one of his random prime numbers as they confront. (Here we send first prime number as the confront i.e. 'a').
Step 8) Here we use public key cryptography algorithm known as RSA.(It uses 2 keys for encryption & Decryption).

2) **Approaches:**



Step 9) Node A send 'a' as they confront to node C.
Step 10) In node C we pertain the RSA algorithm for key.
Generation (public key, private key).
Step 11) Node C sends the public key to node A.
Step 12) Node A encrypts 'a' with the public key of Node C.
Then we get some 'k'. Then node A sends 'k' to node C.
Step 13) Node C receives & decrypts 'k' with C's private key.
Then we get some 'n'.
Step 14) Then node C computes 'c^d mod n' & sends the result to node A. To optimize these computations apply CCM-OKE Model. [Table 1].
Step 15) Now node A sends 'a' as they confront to node B.
Step 16) In node B we apply the RSA algorithm for key generation (public key, private key).
Step 17) Node B sends the public key to node A.
Step 18) Node A encrypts 'a' with the public key of Node B.
Then we get some 'k'. Then node A sends 'k' to node B.
Step 19) Node B receives & decrypts 'k' with B's private key.
Then we get some 'n'.
Step 20) Node B sends 'n' to node C.
Step 21) Node C receives 'n' & computes 'c^d mod n' & sends the result to node A.
Step 22) Since n, c and d are all very large prime numbers, it is unfeasible to determine c and d from the end result of the *mod* function as that is known to be a tough problem.
Step 23) Node A compares both results. If they are equal node B is added to the network. Otherwise discard it [Fig. 3].

In the above SMARUC Algorithm blocks the entrance of malevolent nodes efficiently and the computation of time complexity of a modulo function, reduced with CCM-OKE Model which can be applied to the Diffie-Hellman key exchange algorithm, CRT or RSA key generation. In the above Algorithm from the steps 9 to 21 all the computation time complexity of modular function will be reduced by the following CCM-OKE-Model and then key exchange can be done through Diffie-Hellman key exchange model.

Algorithm for OKE (Optimized Key Exchange) Model to Calculate Pow (w,p) mod n

The Algorithm was modeled based on the binary number system. In which the input parameters are taken in the binary model [10–14].

Steps:

1. Express all the basic input parameters in binary model representation.

   P = (pk−1, pk−2, pk−3 …. p2, p1, p0).

   Where k represents number of bits to represent X.

   W = (wk−1, wk−2, wk−-3, ….. w2, w1, w0)

   N = (nk−1, nk−2, nk−3 …. n2, n1, n0)

2. Assign the value C to 1.
3. Calculate S = pow(w,3) mod n.
4. i = k−1.
5. If i > = 0 then goto step 6 else goto step 11 6. C = mod(C * C, n).
6. If [(pi ! = 0) & (pi-1 ! = 0)] then goto step 8 otherwise goto step 9
7. C = mod(C * C, n); C = mod(C * S, n); i = i−1.
8. If (pi ! = 0) then C = mod(C * w, n)., i = i−1
9. Goto step 6
10. Return the value of C which is the result of optimizes modular function result.
11. END.

The above algorithm reduces the time complexity of algorithm O(log(k)) by reducing the number of multiplications in modular function evolution. Once the Network is established the data among nodes can be transferred or routing tables of networking nodes protected through CCM-OKE Model which is expanded below.

CCM-OKE Model

Character classification [10, 15,16] Model is one simple, efficient with encrypted secured model which is described as.

Steps:

1. Consider a secret key of any length, i.e. one character to any number of characters [Table 1].
2. This secret key is stored in a single dimension matrix called key matrix.

   K = [kl, kZ, ." "" ." ." ." ." ." .. , ku.]

3. Perform XOR operation of each key with RSA-key(Z) generated with OKE model in the above algorithm.

   K={k1 XOR Z, K2 XOR Z, ….. Kn XOR Z}

   Where Z is a value generated optimized Pow(a,d) mod n.

4. Calculate the encrypted Key from key matrix K using the following formula (Phase 1): Kenc = K1®K2®K3… . "…….. ®CKn where ® iS XOR operator.
5. The random Number for performing shift operations on obtaining kenc key obtained by applying rough sets classification Model on Combined set for Plain Text ASCII values and Secret Key ASCII values.
6. TakeS1 = {Set of ascii values of plain text}

   S2 = {Set of ascii values of secret key}

   S = S1 U S2

   Let us example S={78, 66, 69, 75, 101, 114, 121, 21, 112, 8, 110, 34, 56, 23, 10, 66, 77, 89, 119, 114, 123, 147}

   Then classify the sets [15, 16].

Let us suppose rnd value is 66 which

M1 = { > rnd} = {77,89,119,114 …} or {upper approximation values}

M2 = {< rnd} = {6,8,34,77,75 …} or {lower approximation values}

M3 = {== rnd} = {66} or {equivalent approximates values}

S = MAX(M1, M2, M3).

7. Calculate the encrypted key Kenc, by following way (Phase 2):

   KEnc = KEnc >> S. so S number of times perform shifts.

8. Finding this encrypted key is grounded on ASCII value of each and every character of secret key.
9. Plain text is stored in single dimension matrix called PT.

   For example: PT[] = ABCDEFGHIJKLMNOP.....

10. It is grounded on ASCII value of encrypted key.

    C: r[f] = PT[i] Kenc

11. Add both KenC and plain text character by character where i = 1,2, … ,n characters of plain text.
12. Got the Cipher text in matrix CT.
13. That CT sends to the open SSL for double encryption to get two-level encrypted cipher text [Tables 2, 3 and 4].
14. Calculate the reverse process to get back plain text again.

1.3 Open SSL

The open ssl (Secure Socket Layer) unit provides a amalgam multiplicity of commands each of which often has a prosperity of options and wiles (Fig. 2).

The pseudo-commands list-standard-commands, listmessage-digest-commands, and list-cipher-commands output a list of the names of all standard commands, message digest commands, or cipher commands, respectively, that are available in the present open ssl utility.

enc.sh

```
#$1 = filename
Open ssl enc -aes-256-cbc -salt -in $1 -out y.txt -pass
#pass:"sirkrs123"
```

dec.sh

```
Open ssl enc -d -aes-256-cbc -in y.txt -out y.dec -pass
#pass:sirkrs123
```

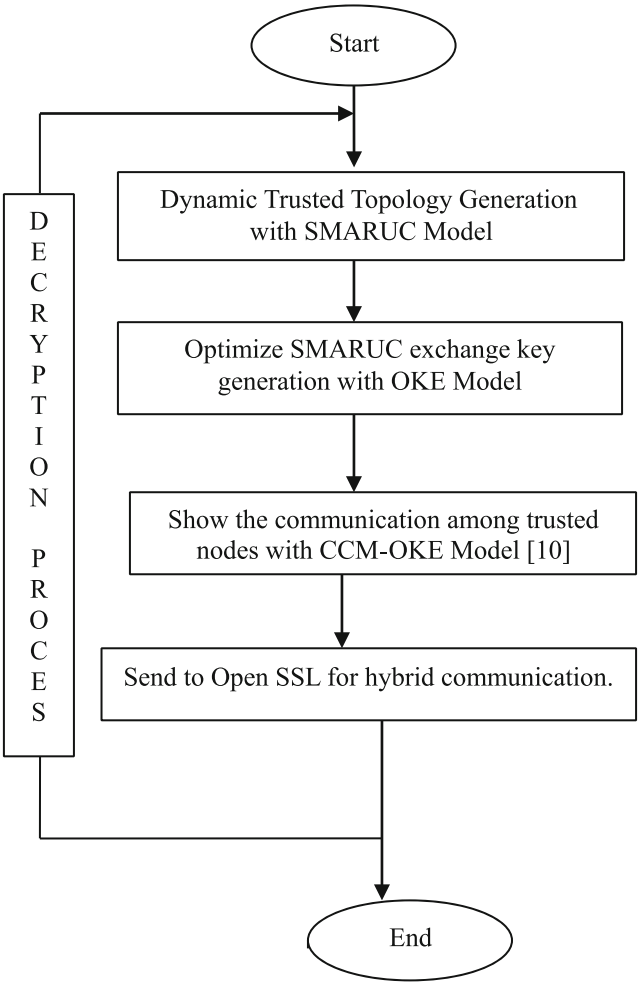


Fig. 2. Flow diagram of SMARUC-CCM-OKE Model

2 Simulation Analysis

The Simulation analysis was performed using NS2 Simulator Package [25].

A. *Simulation Parameters applied on analysis tool are*

- Simulation time: 100 s □ Mobility: Random Way Point.
- Network layer: IP 127.0.0.1
- Source Node: 1
- Destination Node: 2
- Protocol: AODV [3] □ Minimum Speed: 0 m/s □ Maximum Speed: 20 m/s.
- Mac Type: 802.11
- Link Layer Type: LL

- Interface Type: Queue □ Traffic type: CBR
- Packet Size: 28,38,48.

Once the network is established with smaruc model for showing the communication of data from one to another node with CCM-OKE model let us suppose Table [1].

Input data : " Simulator tool "
Secret Key : "RscOkeM"

3 Experimental Results

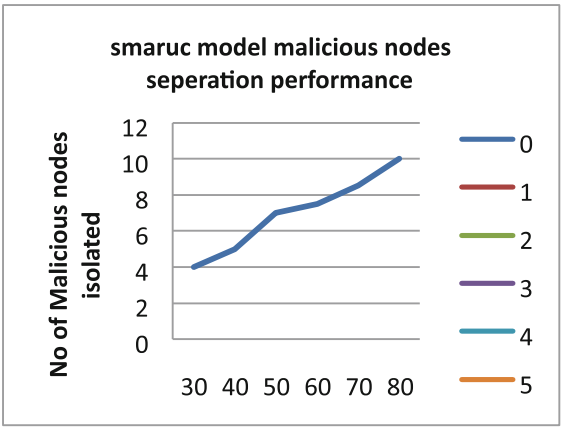


Fig. 3. Malicious Nodes Separation.

Time Comparison Results Between OKE and Without OKE

Performance Comparison Results of Symmetric classical models with proposed Model

Where entropy defines the dissimilarity index of cipher text and Floating frequency defines the maximum repeated character Description.

Performance Results of symmetric Modern Algorithms with proposed Model (Table 5).

Table 1. Characters with its ASCII values

S.NO	Input Text/Secret Key	ASCII
1	S	**115**
2	**I**	**73**
3	**M**	**87**
4	**U**	**89**
5	L	**83**
6	A	**65**
7	T	**94**
8	O	**76**
9	R	**85**
11	T	**80**
12	O	**86**
13	L	**83**

Table 2. CCM-OKE Model Cryptanalysis Values

S.NO	Text Length	Plain Text	Random range	Cypher text	Kenc
1	12	cryptography	10	$%^ww##	30
2	14	Journal paper	20	&&# @We!&67	10
3	8	criticism	15	4rrty6134	20
4	15	Simulation model	10	Qqsr4%%^*	14

Table 3. Analysis of symmetric classical algorithms with proposed Model

S.NO	Encryption Algorithm	Plain Text	Cipher text	Entropy Value	Floating Frequency(MAX)
1	Optimize d CCMOKE Model	Forming encrypted textt	&$et%rtwqwe5!*tuuy	4.42	3
2	Vigenere (B)	Forming encrypted textt	Fptpmsm lvbjbgss jvpmno	3.44	3.84
3	Play Fair (C)	Forming encrypted textt	EPHFKBLFADSWQUMNOLY QQY	3.97	3
4	XOR (D)	Forming encrypted textt	S20Eac/OSA-)1d	0	0
5	Solitaire (E)	Forming encrypted textt	JLBKQMYKRJLLIBUYOWVDT	3.78	

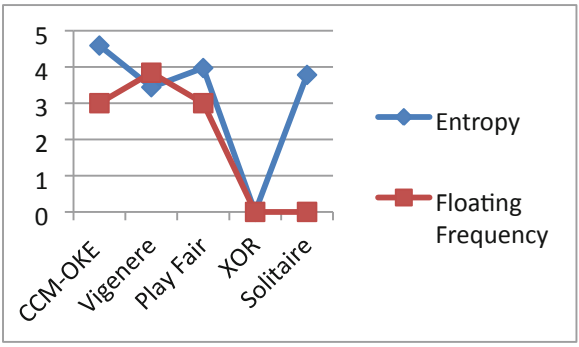


Fig. 4. Quality of cipher in terms of entropy and floating frequency among symmetric classical models

Table 4. Analysis of symmetric Modern algorithms with proposed Model In terms of entropy and frequency

S.NO	Encryption Algorithm	Plain Text	Cipher Text	Entropy Value	Randomness Test
1	Optimized CCMOKE Model	Analysys Of Symmetric Modern algs	$$R%WE2!(gg ©SQcj@#xcvdzSs##&p	5.43	2.56
2	IDEA	Analysys of Symmetric modern algs	x.>.. >..ag ~..u3..[j*c..M|JY	5.12	0.20
3	RC4	Analysys of symmetric modern algs	K!.4..t.F..T.H..o.NTC..j..t*BQ	4.74	0.24
4	DES	Analysys of Symmetric modern algs	..h7.. + l#HJ…w.}.3..k&..Gx… = .	5.22	0.31
5	TRIPPLEDES	Analysys of Symmetric modern algs	H….U..l…m..’../le..\i.z.2E..l	5.27	1.25
6	AES	Analysys of symmetric modern algs	. < …..{…9’.}..dT{[g..N).p…”(.4	5.41	0.37

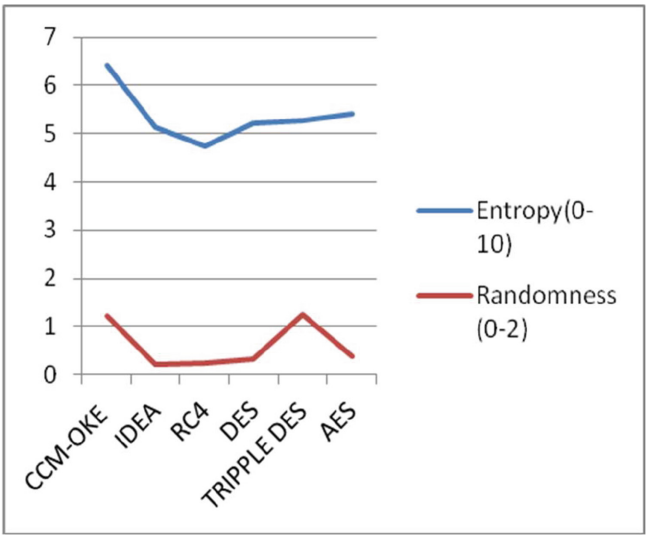


Fig. 5. Performance Comparison of asymmetric Algorithms With proposed Model

Table 5. Analysis of entropy and randomness of Asymmetric Modern algorithms with proposed model.

S.NO	Encryption Algorithm	Plain Text	Cipher text	Entropy Value	Randomness Test
1	Optimized CCM-OKE Model	Analysys of symmetric modern algs	l$@@WQ6!(FF©Sq2j@#xc3erzWs &&y	7.43	1.21
2	RSA	Analysys of symmetric modern algs	.f….Eiv.Y…. =.(< …?Z…’.H..v!…E ….’R.	5.83	0.19
3	TRIPPLE DES	Analysys of symmetric modern algs	H….U..l…m..’../le..\i.z.2E..l	5.27	1.25

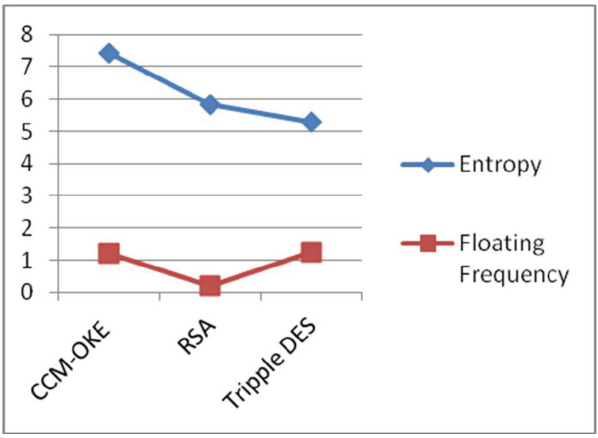


Fig. 6. Analysis of entropy and randomness of Asymmetric Modern algorithms with proposed model.

4 Conclusion

In this paper we have proposed new technique to improve the security in manet in optimizes the modular operation results in terms of both time, space complexities, entropy, Floating frequency [Figs. 4, 5 and 6] and with limited communication cost. The results which we obtained are compared with existed diffie-hellman key exchange, RSA key generation model and various existed crypto algorithms like RSA, DES, Double and tripple DES, AES and so on. The results are verified using crypto tool.

The proposed work specially takes care about initiating the security in MANETs by generating secured dynamic network topology through SMARUC and optimizes the exchange key generation. And it also optimizes the communication cost with reasonable security. The proposed cryptography algorithm developed through user defined dyanamic key . It is observed that which we obtained from the above results, increases the quality of the services among MANETs. In future the above results are comparable with other existed routing protocols and other crypto models for developing better efficient crypto protocol.

References

1. Agrawal, P., Zeng, Q.-A.: Introduction to Wireless and Mobile Systems. Brooks/Cole, Thomson, Pacific Grove, CA (2002)
2. Dhurandher, S.K., Obaidat, M.S.: FACES: friend – based ad hoc routing using challenges to establish security in MANETs systems. IEEE Syst. J. **5**(2), 176–188 (2011)
3. Perkins, C., Royer, E., Das, S.: Ad Hoc on Demand Distance Vector (AODV) Routing. Internet experimental RFC 3561 (2003)
4. Johnson, D.B., Maltz, D.: Dynamic source routing in ad hoc wireless networks. In: Imielinski, T., Korth, H. (eds.) Book Chapter in Mobile Computing, pp. 131–181. Kluwer, Dordrecht, The Netherlands (1996)

5. Jiang, M., Li, J., Tay, Y.C.: Cluster Based Routing Protocol (CBRP) Functional Specification (1998)
6. Sanzgiri, K., Levine, B.N., Shields, C., Dahill, B., Belding-Royer, E.M.: A secure routing protocol for ad hoc networks. In: Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP), Paris, France, pp. 78–89, 12–15 November 2002
7. Hu, Y., Perrig, A., Johnson, D.B.: Ariadne: a secure on-demand routing protocol for ad hoc networks. Wirel. Netw. **11**(1–2), 21–38 (2005)
8. Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: Proceedings of the MobiCom 2000, Boston, MA, pp. 255–265 (2000)
9. Chiang, C.C., et al.: Routing in clustered multihop, mobile wireless networks with fading channel. In: Proceedings of the IEEE SICON'97 (1997)
10. Hasan, A., Sharma, N.: New method towards encryption schemes. In: International Conference on Reliability, Optimization and Information Technology (2014)
11. Vollala, S., Varadhan, V.V.: Efficient modular multiplication algorithms for public key cryptography. In: 2014 IEEE International Advance Computing Conference (IACC)
12. Desmedt, Y.: ElGamal public key encryption. In: Encyclopedia of Cryptography and Security, p. 183. Springer, US (2005)
13. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120126 (1978)
14. Agnew, G.B., Mullin, R.C., Vanstone, S.A.: Fast Exponentiation in GF (2n). In: Advances in Cryptology—EUROCRYPT 1988, pp.225–231. Springer Berlin Heidelberg (1988)
15. Syeda, M., Zbang, Y.Q., Pan, Y.: Parallel granular neural networks for fast credit card fraud detection. In: Proceedings of the IEEE International Conference on Fuzzy Systems 2002, vol.1, pp.572–577 (2002)
16. Polkowski, L., Kacprzyk, J., Skowron, A.: Rough Sets in Knowledge Discovery 2: Applications, Case Studies, and Software Systems. Physica-Verlag, Heidelberg (1998)