Multi-Service Cryptographic Scheme for Secure Data Communication

Marghny H. Mohamed,

Faculty of Computers and Information Assuit University Egypt

Summary
Satisfying security services presents the main challenge for security scheme. In this paper, a new cryptosystem based on hybrid approaches is proposed to provide multi-security services. The proposed method takes the advantage of neural networks computation power to serve data confidentiality. Where using of MAC technique provides authentication and the integrity of this scheme. The proposed scheme is accomplished through three stages, the key generation, encryption stage and the decryption stage at communication ends. This scheme can be used with several types of data: text, image, and sound, and will provide confidentiality, integrity and authentication which is considered as one of the important security services in most applications.

Key words:
Security Services, Neural Network, Cryptography, Message Authentication Code (MAC).

1. Introduction

The massive use of the communication networks for various purposes in the past few years has posed new serious security threats and increased the potential damage that violations may cause. The main challenge in data communication is in keeping data secure against unlawful interference, some of the common serious attacks which threaten data security today are: interception; which occurs when an unauthorized party gains an access to read protected file, fabrication; where an unauthorized party claims to be authorized, and modification where an unauthorized party might be able to modify the transmitted data.

This paper provides a new cryptosystem based on hybrid approaches to support multiple security services such as: confidentiality, to ensure that information is accessible only to those authorized to have an access, authentication, to ensure that a message is coming from the source from which it claims to come, and integrity, to ensure that data is not accidentally or deliberately modified in transit. These services are considered as one of the most important security services required in data transmission through insecure channels.

Cryptosystems rely on the assumption that a number of mathematical problems are computationally intractable in the sense that they cannot be solved in polynomial time. Numerous approaches have been
applied to address these problems. In the proposed scheme, the confidentiality service is accomplished by designing encryption algorithm based on neural networks mechanism, integrity service is provided by using message authentication code (MAC) approach, and the authentication service is done by using a digital signature technique based on RSA encryption algorithm.

Artificial neural networks (ANNs) are relatively new computational tools that have found extensive utilization in solving many complex real-world problems. The attractiveness of ANNs come from their remarkable information processing characteristics pertinent mainly to nonlinearity, high parallelism, fault, noise tolerance, learning and generalization capabilities.

Neural networks have been attracting more researchers since the past decades. The properties, such as parameter sensitivity, random similarity, learning ability, etc., make it suitable for information protection, such as data encryption, data authentication, intrusion detection, etc. [1]. According to the properties of random similarity and parameter sensitivity, neural network has been used in data encryption such as [2, 3, 4].

The one-way property makes neural network a suitable choice for hash function [5]. Hash function is a technique for data integrity, which takes a message of arbitrary length as input and produces an output of fixed length. The hash value is often much shorter than the message, which makes it suitable for digital signature or data authentication, as a hash function. It should be easy to compute the hash value from the message, while difficult to compute the message from the hash value, this property is called one-way property. According to this case, some hash functions based on neural networks have been presented such as [6, 7, 8, 18].

ANNs also used in another data security issues, such as, authentication [1, 9, 10], handprint identification [11], recognition and classification of computer virus attack [12], intrusion detection [19], data masking [20], etc.

Traditional cryptographic algorithms, such as DES, AES, RSA, etc.[13, 14] send the ciphertext over the cyber-space while keeping a secret part (i.e. key) shared, which tend to be dangerous, as any intruder

Manuscript received July 5, 2011
Manuscript revised July 20, 2011

can get the encrypted message and apply his own cryptanalysis techniques. this meaning when the data. This means that travels over the network even though it is hidden more attacks could be applied to the cipher message trying to get full or partial information from the message, compared to our scheme where less number of such attacks can be applied, as the message is hidden with the sender. In our scheme, the data sent over the communication channel is not the encrypted message, but a neural network model (weights) is generated by the sender. at the receiving end, the receiver generates the message with the help of the model received, along with secret shared information between the two parties. Also, sending the digital signature of the original message along with the cipher enables the receiver party to verify the sender identity and message integrity.

The rest of this paper is organized as follows: section 2 introduces background of the basic concepts used in this work, the proposed scheme is presented in section 3, experimental results are presented in section 4, in section 5 the security analysis is introduced, finally conclusions are provided in section 6.

2. Background

In the world of communications, assurance is sought that:

1. The message is protected against unauthorized individuals reading information that is supposed to be kept private.
2. A message is not accidentally or deliberately modified in transit by replacement, insertion, or deletion.
3. The message is coming from the source from which it claims to come.

Our model is based on several mechanisms like, artificial neural networks (ANN), message authentication codes (MAC), RSA encryption algorithm.

To build the encryption algorithm in our scheme we used radial basis function neural networks.

2.1 Radial Basis Function Networks

A Radial Basis Function Network (RBF) is an artificial neural network that uses radial basis functions as activation functions. RBF networks are powerful techniques for interpolation in multidimensional space. It provides an attractive approach for function approximation because of its flexible structure, fast training, powerful generalization capability, and conceptual elegance [15].

The basic architecture of a RBF network is shown in Fig.1. It includes three layers: the input layer, hidden layer and output layer. The nodes within each layer are fully connected to the previous layer as shown.

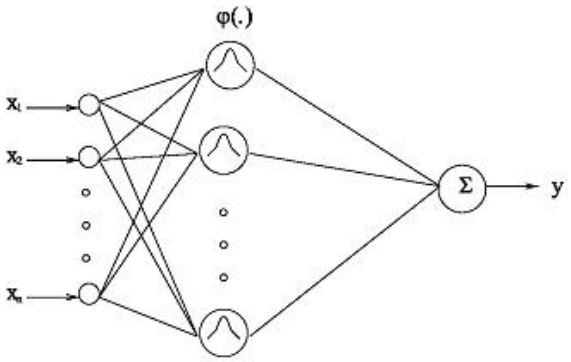

Fig .1: A radial basis function network

The input variables are assigned to each node in the input layer and are passed directly to the hidden layer without weights. The transformation from the input layer to the hidden layer is nonlinear, whereas the transformation from the hidden layer to the output layer is linear.

An activation function for a hidden layer node is a radial basis functions.

The output of the network can be expressed as:

$$f(x) = w_0 + \sum_{i=1}^{m} w_i \phi_i(||x - c_i||) \quad (1)$$

where x is an input pattern, ci is the center for hidden node i, wi is the weight between hidden node i and the output node, and w0 is a bias weight, m is the number of hidden nodes, $\phi_i(.)$ is the activation function for the hidden layer. The norm ||.|| is typically taken to be Euclidian distance.

And the basis function is taken to be Gaussian functions, which can be expressed as:

$$\phi_i(u) = e^{\frac{-u^2}{2\sigma_i^2}} \quad (2)$$

Where σ_i is the width of hidden layer neuron i, it is used to control the spread of the RBF.

The output of the RBF network can be expressed as:

$$f(x) = w_0 + \sum_{i=1}^{m} w_i e^{\frac{||x - c_i||^2}{2\sigma_i^2}} \quad (3)$$

2.2 Message Authentication Codes (MAC)

A MAC is a technique which involves the use of a secret key to generate a small fixed-size block of data, known as a cryptographic checksum or MAC that is appended to the message. It provides a way to check the integrity of transmitted information among communications parties. For our demonstration we used CBC-MAC mechanism [16], as shown in Fig.2.

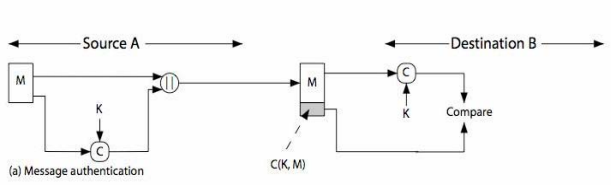

Fig. 2: The use of MAC

RSA is a public-key encryption algorithm [14], used here to sign the MAC value by encrypting it with the private key of the sender, the combination of public key encryption and MAC technique formed mechanism to provide the digital signature of the proposed scheme.
A digital signature is defined as an encrypted message digest, by the private key of the sender, appended to the message which is then sent to the receiver. Digital signature enables the recipient of information to verify the authenticity of the information origin, and also verify that the information is intact. Thus, it provides authentication and data integrity.

3. The Proposed Scheme

The proposed scheme can be summarized in the following stages:
At *sender side* some stages must be done:

- **Key Generation Stage:**

At this stage four kinds of secret keys are generated and shared among the communication parties, as shown below:

1. Using RSA algorithm, two keys are generated for every user of the communication, private key (d, n1), which is exclusively confined to the user, and a public key (e, n1), which is published among other members of communication.
2. Another key kmac, is generated to be used in computing MAC value of the original message.
3. Finally, another key krbf, is assigned to be used in the encryption algorithm (neural network model).

- **Producing Digital Signature Stage:**

This stage consists of two phases, first one is done by computing CBC-MAC value of the message M after divided it into a set of blocks with a fixed size (m).

$$M = \{ b_1, b_2, .., b_n \}. \quad (4)$$

Where n is number of the blocks. Every block consists of m elements.

$$b_i = \{ x_1, x_2, .., x_m \}. \quad (5)$$

Secondly sign the MAC value by encrypting it by the sender's private key (d, n1), to produce signed MAC (MAC'):

$$MAC' = MAC^{\,d} \ mod\ n1. \quad (6)$$

- **Encryption Stage:**

The blocks are encrypted as follows:
For every block b_i in the message train the RBF neural network model, by putting b_i as target and the key k_{rbf} as input to this model, after training is accomplished, the generated set of weights and biases values from the neural network model are kept every time.

$$W = \{ w_1, w_2, .. , w_n \} \quad (7)$$

By generation the values of (W, MAC') this stage goes to an end, and these values are sent as bunch to the receiver(s). Fig. 3, shows the steps at sender side.

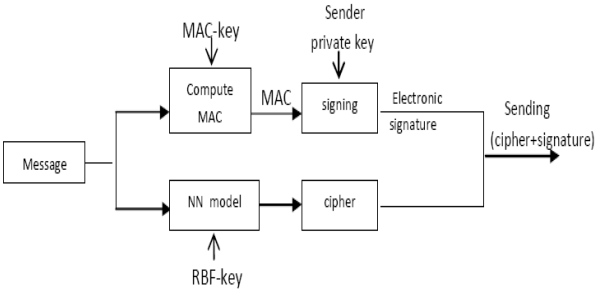

Fig.3: The block diagram of steps at sender side

At the receiver side another steps are done to decrypt the ciphertext and retrieve the original message, in addition to ensuring authenticity and message integrity according to the following stages:

- **Decryption Stage:**

1. Using the received weights W as connection weights on the RBF neural network model, and the key k_{rbf} as the input to this model, after training the model, the message blocks will be obtained one after the other as outputs of this model.
2. Construct the message M from the obtained blocks, as in equation (4).

- **Verification Stage:**

1. Compute *MAC* value of the obtained message (*Obt-MAC*).
2. Decrypt the received signed *MAC* value *MAC'* by sender's public key for verifying identity of the sender:

$$MAC = (MAC')^{e} \ mod\ n1 \quad (8)$$

3. Compare the obtained MAC value from equation (8) with the MAC value of the constructed message *Obt-MAC*, if the matching obtained (*MAC = Obt-MAC*). This indicates that the

message is not modified in transmission among the communication parties. Fig.4 shows the steps at receiver side.

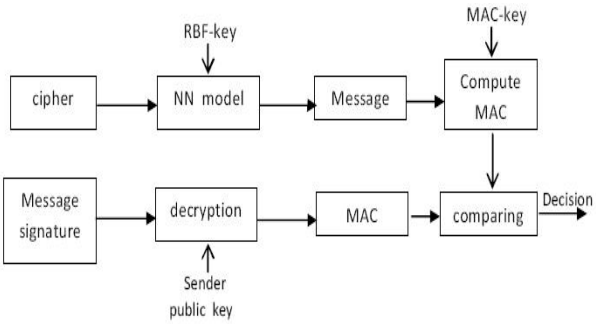

Fig.4: The block diagram of steps at receiver side

4. Experimental Results

In order to evaluate the effectiveness of the proposed scheme, the following experiment has been conducted to measure the level of confusion and diffusion, by comparing plain to cipher relationship as a metric model for security. These simulation experiments have been done on a sentence M representing the original message:

M = "Cryptographist is the science of overt secret writing", to encrypt this message by the proposed model, we used it as a target to the neural network model, and the key *k* as the input. Suppose k_{rbf} is "abcde".

The resulted ciphertext was: {-220.483, -113.4, -133.322, -128.663, -167.667, -156.011, -88.6256, -76.2291, -135.732, -63.1311, -127.282, -71.9748, -142.11, 0, -240.736, -66.4513, -12.9577, -286.048, 53.4062, -223.783, 43.063, -303.877, 48.2367, -194.677, -36.2589, -154.819, -38.8569, -174.476, 4.7563, -256.29, -29.8318, -65.3878, -233.741, -14.6295, -129.448, -75.0061, -150.272, 23.1875, -289.575, 26.002, -154.022, -105.869, -47.6026, -189.114, 46.5301, -301.499, 30.315, -146.859, -94.1419, -72.9435, -136.546, -41.3305, -184.011}.

The contrast between plaintext and ciphertext is demonstrated in Fig.5.

From the resulted ciphertext we can clearly notice that for each character on the original message there is a different value appeared in the ciphertext, and there is no direct relationship between the plaintext (output of model) and the cipher text (weights). This indicates that the proposed model has a high confusion, because the relationship between the input (key) and the output (message) is nonlinear as it depends on the nature of neural network.

Also we observed that the message has some repeated characters such as character "*e*" for example (repeated 6 times), and every time the resulted cipher is different from the other, the repeated values are disappeared on the resulted ciphertext. Fig.6 shows the distribution of the "e" character in the ciphertext. This indicates that the proposed model provides a high level of diffusion.

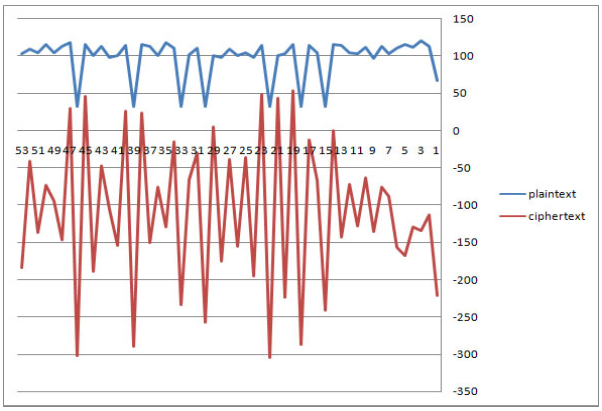

Fig.5: The Contrast between Plaintext and Ciphertext

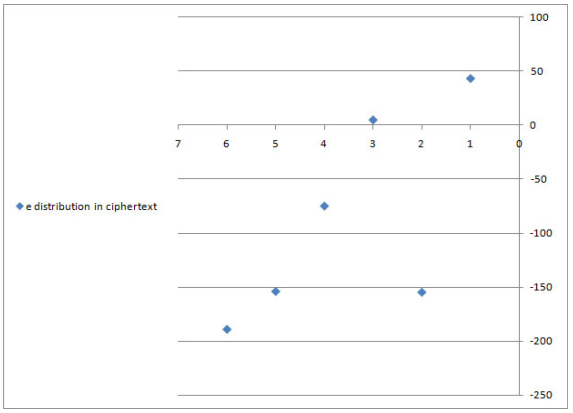

Fig.6: Distribution of character 'e' on ciphertext

A similar experiment has also been conducted to a sentence consisting of consecutive m's as a plaintext with the length=10, and the same key as the previous experiment, the resulted ciphertext was: {0, 0.9589, 0.7183, 0.7183, 0.9589, 0.9619, 0.263, 0.7581, 0.263, 0.9619}.

Even under this very extreme condition, no relation between plaintext and ciphertext can be noticed, and the distribution of ciphertext is random. This confirms what we mentioned about the confusion and diffusion properties is provided by the proposed scheme.

To confirm our results one more experiment is conducted. We encrypted another message similar to the previous one using the same key used before, to see what happens when two very similar texts are encrypted under the same key.

Suppose *M = "mmmnmmmomm"*.

The resulted ciphertext is: {0, 0.0892, -0.2902, 1.1603, -0.2678, 1.0292, -2.5630, 4.4555, -2.5630, 1.0292}.

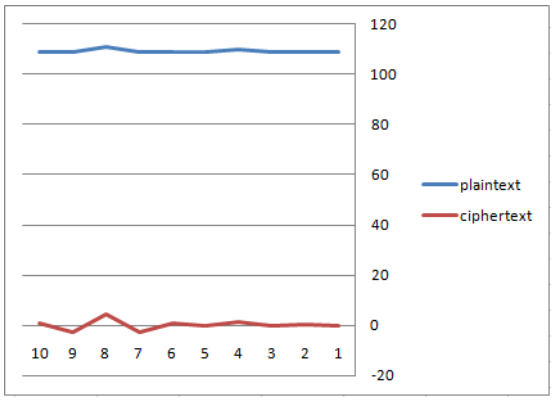

Fig.7: The Contrast between Plaintext and Ciphertext

As we can see clearly, the resulted ciphertext is completely different from the previous ciphertext although the two messages are same. The contrast between plaintext and ciphertext is demonstrated in Fig.7.

Another experiment is performing on the previous message using a similar key.

M = *"mmmnmmmomm"*, k_{rbf} = *"aacde"*, the resulted ciphertext is:

{-0.8200, -0.7501, -1.2799, 0.4468, -1.3067, 0.0000, -2.8444, 3.6444, -2.8444, 0}.

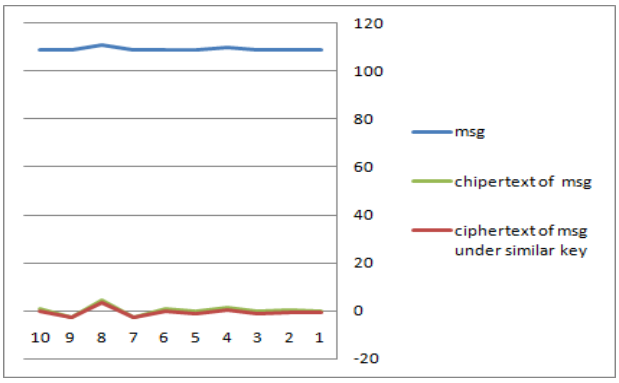

Fig.8: The Contrast between the same message and its Ciphertext under similar keys

Again we can see, the resulted cipher is totally different from the previous experiment as shown in Fig.8, although it is the same message and encrypted under similar key.

5. Security Analysis

Some security analysis has been performed on the proposed encryption scheme, such as:

- *Known-plaintext* attack: Suppose the intruder knows some pairs of ciphertexts and corresponding plaintexts, here his goal is to reveal the shared data (keys), to use it in future for decipher other ciphertext. The intruder will then have to search in a semi-impossible search space. Consider the ANN model in the proposed scheme; the attacker must firstly construct the set of possible key space. If we suppose the key size of the neural network model is $p \times q$, and each unique cell in the matrix takes 2^x, where x is depending on the data type used, so such complexity can be randomly measured as:

$$Z = (2^x)^{p \times q} \quad (9)$$

This value considered as the possibility of finding the key (inputs) of the neural network model. Due to the non-linearity of ANN used by our model, the possibility of predicting the input is not unique as several inputs can be mapped, in addition to the one way property of neural network makes computing the key of it (inputs) difficult enough even if the intruder knows the ciphertext (weights) and the plaintext (outputs).

- Ciphertext-only attack: Suppose the intruder can eavesdrop the ciphertext in transmit, his goal here is to reveal the keys (inputs), or the corresponding plaintext (outputs), then the intruder must search in the key space of inputs (Z), and in the key space of outputs ($Z \times n$), where n is the number of message blocks, then the intruder must search in the following key space: $Z^2 \times n$, the bigger Z and n are the larger the key space which search on it.

Also for ANN weights with unknown inputs and outputs is meaningless as weights can map X inputs to Y outputs without detecting which one is the target pair.

- *Confusion and diffusion*: Confusion and diffusion are two basic design criteria for encryption algorithms [17]. Diffusion means spreading out the influence of a single plaintext symbol over many ciphertext symbols so as to hide the statistical structure of the plaintext. Confusion means the use of transformations to complicate the dependence of the statistics of ciphertext on that of the plaintext. In the proposed algorithm, we can detect no such formulation to predict the behavior of such distribution. Therefore the proposed cryptosystem has a high confusion and diffusion properties, which makes the cryptosystem of high key sensitivity and plaintext sensitivity, and thus of high computing security. On the other hand, the mapping function of the used neural network model is a nonlinear function which makes the relationship between the plaintext, key and ciphertext nonlinear. This property complicates possibility of retrieving one of them even the others were known.

6. Conclusion

This paper is proposed a new cryptosystem based on hybrid approaches proposed. The proposed cryptosystem provided multi-security services such as confidentiality, authentication, and integrity, which are considered as one of the important security services in most applications. To serve data confidentiality, an encryption algorithm based on neural network is proposed, whereas the combination between MAC technique and RSA encryption algorithm produced the digital signature of the scheme for providing the other mentioned security services. In this scheme the digital signature firstly produced by computing the MAC of the message, then signing it by the sender's private key generated by RSA algorithm. The encoding of the message is done by using neural network, and shard key as input to the model, and the data need to be secured will be the output of it, and the generated set of weights will be act as the cipher. Finally, the signed MAC and cipher will be sending to cyber-space. At the receiver end, after the decryption is done, the digital signature can be used to verify the integrity of the message, and the authentication of the sender. The experimental results indicate that the proposed cryptosystem has a high confusion and diffusion properties, therefore it has high security and it is suitable for secure communications.

References

[1] Shiguo L.: "Image authentication based on neural networks", *CoRR*, 2007.

[2] Shiguo L.: "A block cipher based on chaotic neural networks", *Neuro-computing*, vol. 72[4-6]: pp. 1296 - 1301, 2009.

[3] Khalil S.: "A backpropagation neural network for computer network security", *Journal of Computer Science*, vol. 2[9]: pp. 710 - 715, 2006.

[4] Wenwu Y. and Jinde C.: "Cryptography based on delayed chaotic neural networks", *Physics Letters A*, vol. 356[4-5]: pp. 333 - 338, 2006.

[5] National institute of standards and technology: "*FIPS 180-2: Secure Hash Standard, Federal Information Processing Standard (FIPS), Publication 180-2*", 2002.

[6] Shiguo L., Jinsheng S., and Zhiquan W.: "Secure hash function based on neural network", *Neurocomputing*, vol. 69[16-18]: pp. 2346 - 2350, 2006.

[7] Di X., Xiaofeng L. and Yong W.: "Parallel keyed hash function construction based on chaotic neural network", *Neurocomputing*, vol. 72[10-12]: pp. 2288 - 2296, 2009.

[8] Pengcheng W., Wei Z., Huaqian Y. and Jun C.: "Combining rbf neural network and chaotic map to construct hash function", *Springer Berlin/Heidelberg*, vol. 3973: pp. 332 - 339, 2006.

[9] Tiegang G., Qiaolun G. and Sabu E.: "A novel image authentication scheme based on hyperchaotic cell neural network", *Chaos, Solitons & Fractals*, 2009.

[10] Shahbaz Z. and Mehregan M.: "User Authentication Using Neural Network in Smart Home Networks", *International Journal of Smart Home*, vol. 1[2]: pp. 147 - 154, 2007.

[11] Jun K., Yinghua L., Shuhua W., Miao Q. and Hongzhi L.: "A two stage neural network-based personal identification system using handprint", *Neurocomputing*, vol. 71[4-6]: pp. 641 - 647, 2008.

[12] Anastasia D., Konstantinos M., Dimitris G. and Sokratis K.: "Design of a neural network for recognition and classification of computer viruses", *Computers & Security*, vol. 14[5]: pp. 435 - 448, 1995.

[13] William S.: "*Cryptography and Network Security Principles and Practices*". Prentice Hall, fourth edition, 2005.

[14] Bruce S.: "*Applied Cryptography: Protocols, Algorithms, and Source Code in C*". Wiley Computer Publishing, John Wiley & Sons, Inc., second edition, 1995.

[15] Mark J.:"Introduction to Radial Basis Function Network". University of Edinburgh, 1996.

[16] International Standard, ISO/IEC 9797-2: "*Information technology-Security techniques-Message Authentication Codes (MACs)-Part 2: Mechanisms using a dedicated hash-function*", 2002.

[17] Shannon C.: "Communication theory of secrecy systems". *Bell Systems Technical Journal*, vol. 28: pp. 656 - 715, 1949.

[18] Zhongquan H.: " A more secure parallel keyed hash function based on chaotic neural network", *Communications in Nonlinear Science and Numerical Simulation*, vol. 16[8]: pp. 3245-3256, 2011.

[19] Gang W., Jinxing H., Jian M., and Lihua H.: "A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering", *Expert Systems with Applications*, vol. 37[9]: pp. 6225-6232, 2010.

[20] Vishal G. and Ashutosh S.: "A neural network approach for data masking", *Neurocomputing*, vol. 74[9]: pp. 1497-1501, 2011.

Marghny H. Mohamed received his Ph.D. degree in computer science from the University of Kyushu, Japan, in 2001, his M.Sc. and B.Sc. from Asyut university, Asyut, Egypt, in 1993 and 1988, respectively. He is currently an associate professor in the Department of Computer Science, and Vice-President for Community Services and Environmental Affairs of the Faculty of Computers and Information Systems, University of Asyut, Egypt. His research interests include data mining, text mining, information retrieval, web mining, machine learning, pattern recognition, neural networks, evolutionary computation, fuzzy systems, and information security. Dr. Marghny is a member of the Egyptian mathematical society and Egyptian syndicate of scientific professions. He is a manager of some advanced research projects in Faculty of Computers and Information Systems, University of Asyut, Egypt.