International Journal of Security and Its Applications
Vol. 11, No. 9 (2017), pp.13-22
http://dx.doi.org/10.14257/ijsia.2017.11.9.02

Cryptography Protocol: A Novel Multilingual Adaptive Encryption Technique with Phonetic Based Ciphering

Ahmed Mokhtar A. Mansour[1] and Mona A. M. Fouad[2]

[1]CTO, Nile Innovations, Cairo, Egypt
[2]National Telecommunications Institute, Cairo, Egypt
[1]ahmedmokhtar_adu@yahoo.com, [2]mfouad@nti.sci.eg

Abstract

This paper proposes a novel hybrid encryption algorithm that utilizes the natural language phonetics significantly. The encryption process is based on two major aspects. The first is the phonetic difference between languages and their mapping into Human Machine interface tools such as the computer keyboard. The second aspect is the embedded ciphering process as a mean of private key exchange (PKE). Such a key will be used to determine all the ciphering parameters. The presented encryption protocol is an end-to-end process and the choices of languages and machine interface is completely user dependant. The creation of ciphering parameters such as the indicator, the key, the array and the operation are also user defined. This work explores several examples for each choice using the Arabic language as a model for the phonetic intermediate language. The keyboard character mapping is used based on the location of characters in the multilingual keyboard. Both fixed location mapping and variable location shifting are introduced.

Keywords: *Information security, encryption, Private Key Exchange (PKE), ciphering, Natural Language Processing (NLP)*

1. Introduction

Modern communication systems support Homomorphic encryption scheme, which performs processing directly on the encrypted data [1]-[3]. This new orientation opened the door for further development and enhancement of data encryption and security [4]–[8]. The tradeoff between complexity and security level is an important issue that is the main objective of the proposed encryption scheme.

This work introduces a novel hybrid encryption algorithm. The encryption process is composed of two main parts, namely; the phonetic multilingual encryption and the ciphering encryption. The phonetic encryption is based on converting the phonetic of a text into letters from different languages as a first step. After that, the user may reuse the keyboard places (as an example) to map again the letters into the original language resulting in a message of the same language with completely different characters. This work uses the keyboard as an intermediate simple mean of encryption; however, many other intermediates might be used such as the phone board, numbers, etc. The second part of this work will use the encrypted message to perform automatic ciphering and identify the ciphering indicator, the ciphering key, the ciphering array as well as the ciphering operator.

The phonetic based encryption algorithm is applied for both the message and the key. In the first part, an intermediate language is selected to perform the phonetic encryption, and then the encryption process should take place based on the phonetic conversion between the two different languages. After that, the recreation of the message using the

Received (July 13, 2017), Review Result (August 24, 2017), Accepted (August 28, 2017)

ISSN: 1738-9976 IJSIA

same letters of the original language takes place. For example, if the original message is "listen to me my name is Ahmed Mokhtar" – if we choose the Arabic language as intermediate language, the message will be "gsk j, ld lhd kdl hd. H[l] Lojhv". A complete example will be illustrated in Section 3.

The second part is the ciphering process that is also dependant on phonetic encryption. Imagining that the ciphering indicator will be the recipient name after performing the same phonetic ciphering, for example, if the recipient name is "john" the encryption for him would be "[,k" (using Arabic as the intermediate language). In this case, the system will start searching from the end of the message till finding the character '[', which will be the ciphering indicator. All the characters after the ciphering indicator will not be ciphered using the automatic ciphering. However, they might be ciphered in another way. The following characters of the ciphering indicator will be the ciphering array. Notice that the key indicator should be more sophisticated rather than being a fixed data (recipient name).

The ciphering key is generated by applying a certain mathematical operation to the ciphering array before encryption using this key. The mathematical operation depends on the content of the ciphering array. Both the ciphering key and the ciphering array are correlated using another mathematical operation to determine the ciphering index. The ciphering index then will be applied to the message from the beginning till the ciphering indicator. This operation is self contained and completely user defined encryption.

In this work, the intermediate language is Arabic. The ciphering key is the sum of the factorial of the ciphering array, and the operation applied to encrypt the text message is the modulus resulted from applying the ciphering key over the ciphering array elements. The sign of the ciphering key will alternate from positive to negative, and vice verse for each character. The ciphering indicator itself will be produced using the first letter of the encrypted receiver name, as mentioned above. It is important to note that, the un-ciphered part of the message is realized by the user defined characteristic, such as the encrypted receiver name the first letter, the last letter, or the intermediate letter, etc. All the operations are independent of such un-ciphered, yet encrypted, part of the message. Furthermore, another ciphering or encryption operation might be applied to this part, such as the 'A-M' code [9]-[12].

A brief survey of the NLP-based related works exists in Section 2. The detailed steps of the encryption and decryption procedures of the proposed protocol are shown in Section 3. A complete example for applying the developed encryption and decryption processes is illustrated in Section 4. Advantages and challenges as well as the conclusion of the proposed work exist in Section 5 and 6, respectively.

2. NLP-Based Related Works

Both of the cryptography and the steganography aim to secure documents (text or media). Text Secret Sharing Schemes used natural language texts to be the cover media of the transmitted messages, instead of images in steganography [13]-[15].The main challenge of the linguistic steganography is the semantic of the result. Words are substituted by synonyms with the same sense in [13]. A secrete key is embedded into a meaningful text message by distributing its characters, in a certain order, among the words of the exchanged text message [14]. A comparative study of cryptography and steganography, concerning several aspects such as the usage of a secrete key, the carrier type of the secrete message, visibility of the secrete information, types of the applications, etc. is introduced in [15]. Three linguistic transformations of natural languages are performed to encrypt text messages in [16].

Another related NLP-based cryptography is the visual cryptography (VC). A letter based visual cryptography scheme is proposed for hiding multiple secretes into images using natural language letters to perform meaningful text files to represent pixels [17] [18].

Incorporating syntactic and semantic analysis of natural languages, the security policies are automatically extracted in [19]. A two-time pads cryptanalysis algorithm to hack the NLP based cryptography is proposed in [20]. On the contrary, the novel significant utilization of the natural language phonetics strengths the proposed work as exposing the encryption requires perfect recognition of the phonetic interpretation of the intermediate language as well as the existence of its interface.

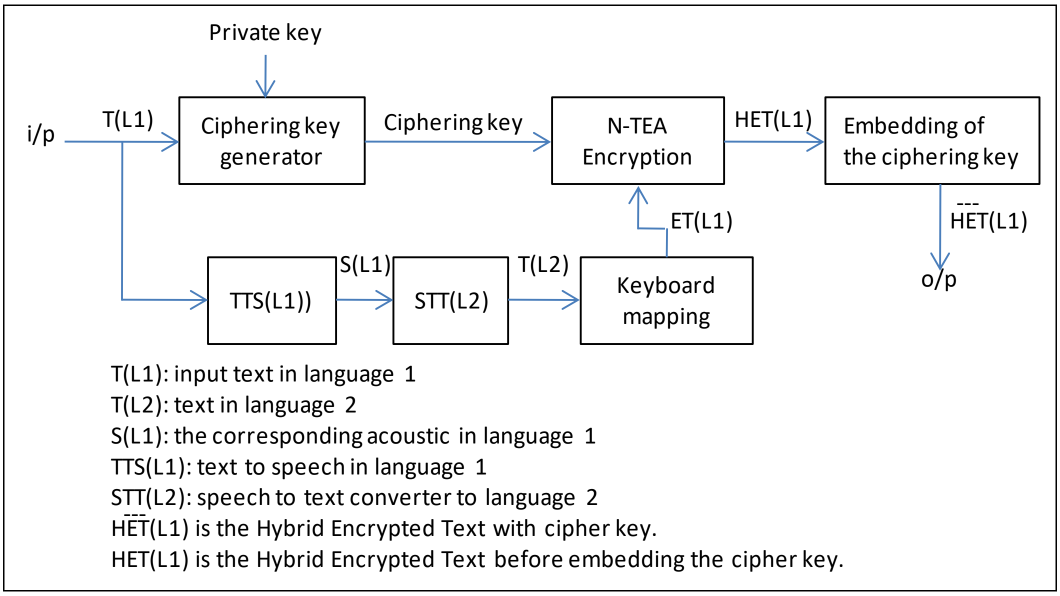
3. The Proposed Protocol

The proposed protocol is phonetic and linguistic based. Each of the encryption and decryption process is composed of two parts; the phonetic multilingual conversion and the ciphering process.

3.1. The Encryption Procedure

As shown in Figure 1, the proposed encryption procedure is composed of five main steps, namely, the generation of the cipher key, the text to speech converter, the speech to text converter, the character mapping, and the ciphering.

1. **Generation of the cipher key:** The private key is used to generate the ciphering key from the original message that will be used in the second step of the protocol. The N-TEA protocol is used for that purpose in the example of section 3, however, the procedure is generic and any other protocol might be used to generate the variable ciphering key. In Figure 1, the input text T(L1) represents the input in the original language of the message and will be forwarded to two different paths. The first path will go to the ciphering generator resulting in the ciphering key. The second path is mentioned in the following three steps.
2. **Text to Speech Conversion:** The original text to be read with an automatic text to speech converter. The purpose is to generate a speech version of the text that will be used in the next step. As mentioned in Figure 1, the T(L1) second path is forwarded to the text to speech TTS(L1) block resulting in S(L1). The S(L1) is a



T(L1): input text in language 1
T(L2): text in language 2
S(L1): the corresponding acoustic in language 1
TTS(L1): text to speech in language 1
STT(L2): speech to text converter to language 2
$\overline{\text{HET}}$(L1) is the Hybrid Encrypted Text with cipher key.
HET(L1) is the Hybrid Encrypted Text before embedding the cipher key.

Figure 1. The Block Diagram of the Encryption Procedure

speech that represents the acoustic(phonetic) representation of the original text T(L1).

3. **Speech to Text Conversion:** The speech to be interpreted with an automatic speech to text converter. This time the speech to text converter will be using another language to write the text. That is to say, the speech message has no meaning in the intermediate language. The converter should convert the phonetics of the speech to text of the chosen intermediate language. This type of message will be letters with no meaning in the intermediate language; however, reading it in the intermediate language allows human who knows the original language to understand the text. Human being may understand up to this point and some intelligent multilingual interpreters can understand as well. In figure 1, the output of the previous step S(L1) continues in the second path to STT(L2) which is the speech to text converter based on the intermediate language (L2). The resulted output T(L2) is a text of the intermediate language L2 as mentioned before.

4. **Character Mapping:** The new text created by the intermediate language is to be mapped to the original message using ASCII code mapping table or any other chosen character mapping table between the two languages, in our case, we chose the keyboard mapping for simplicity for human interaction. The resulting code in this step will be of the original language. The text at this level is neither meaningful for a machine nor for a human. In Figure 1, T(L2) is treated through the keyboard mapping block. This will result in an encrypted message in the original language EM(L1). Note that the keyboard mapping might be as simple as replacing each characters of the intermediate language with the character of the original language that has the same location on the standard keyboard. Complexity might be added such as variable location shift. The variable location shift means that; instead of replacing the character of the original message with the character of the intermediate language in the same keyboard location; the mapping will be done to a shifted character. Shift could be shift right or left with fixed or varying steps. One step shift right for the letter G is the location of letter H and so on.

5. **Ciphering:** In this work, we chose a variable key based ciphering that is created automatically from the original message in case of offline text exchange. The key is based on the N-TEA algorithm that is driven from the 'A-M' code as mentioned in the first step. In figure 1, the results from the different two paths mentioned in step one will be the input for the N-TEA encryption block. That is to say, the ciphering key resulted from step one and the EM(L1) resulted from step four will feed the N-TEA block and will result in a hybrid encrypted text of the original language HET(L1).

The created key in the first step to be hidden in the encrypted and ciphered massage based on the private key, the same private key is used while creating the ciphering key in the previous step. The output of this step is shown in figure 1 as $\overline{HET(L1)}$.

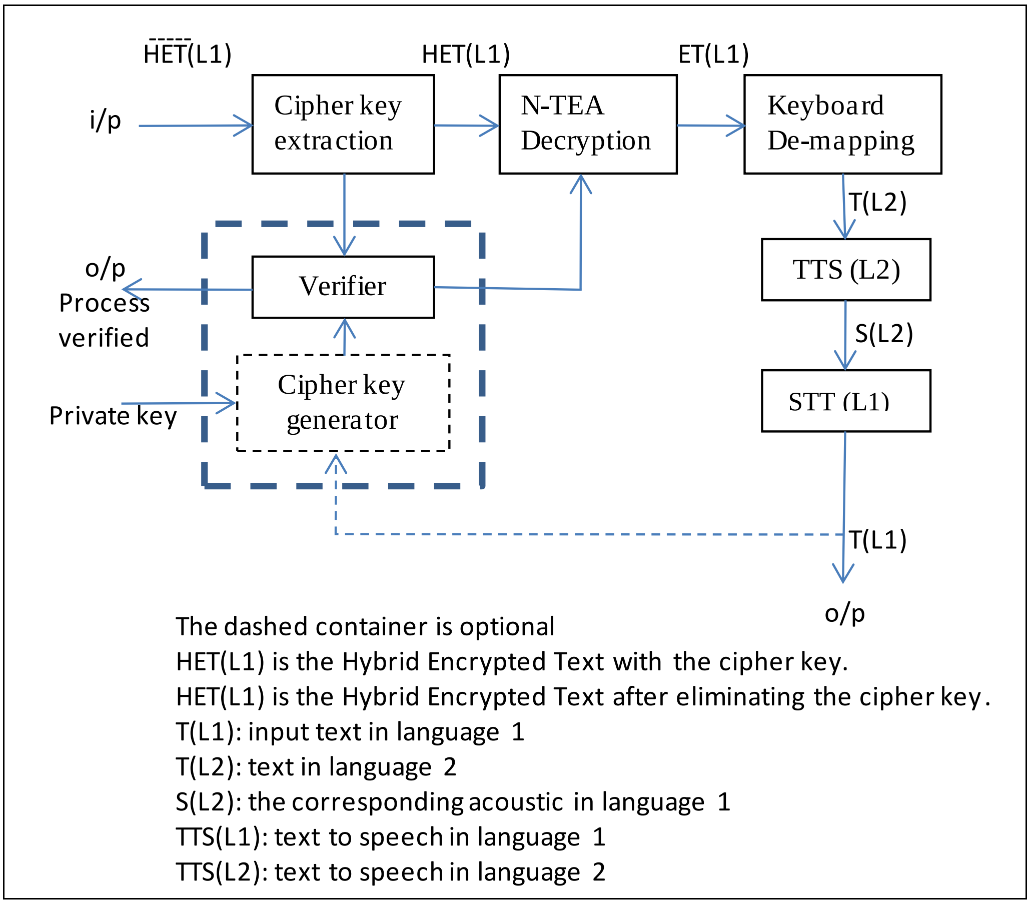
For consistency, the private key itself might be phonetically encrypted as in the first three steps before use to make sure that its content will appear in the encrypted message.

As it might be clear now, the use of speech to text and text to speech converters may affect the accuracy of the message. The reason for that is the meaningless content of the message of the intermediate language.

3.2. The Decryption Procedure

The decryption procedures is exactly the opposite procedure of the encryption one except for the last step where the verification is applied to make sure that the variable ciphering key matches the used one, as shown in Figure 2.

1. **Extraction of the cipher key:** Use the private key to extract the ciphering key from the message and eliminate it. This step will result in two important parts; the first part is the encrypted message without any hidden information inside it, which is ready now to start the decryption process. The second part is the ciphering key to be used to perform the first decryption step as well as performing the verification step at the final stage. In Figure 2, the hybrid encrypted text with the embedded key $\overline{HET(L1)}$ is input of the decryption system. The output of the ciphering key extraction block will be HET(L1) which is the hybrid encrypted text. This text should be identical to the output of the step five in the encryption process. If not, all the following steps will be based on wrong input.



The dashed container is optional
HET(L1) is the Hybrid Encrypted Text with the cipher key.
HET(L1) is the Hybrid Encrypted Text after eliminating the cipher key.
T(L1): input text in language 1
T(L2): text in language 2
S(L2): the corresponding acoustic in language 1
TTS(L1): text to speech in language 1
TTS(L2): text to speech in language 2

Figure 2. The Block Diagram of the Decryption Procedure

2. **Deciphering:** The ciphered (encrypted) message without the ciphering key is decrypted to generate the phonetic based encrypted message. The result of this step is a one level, phonetic-based, decrypted message. In Figure 2, the HET(L1) is decrypted using the N-TEA decryption algorithm. This step will result in a phonetic encrypted text of the original language ET(L1). The output of this step should be identical to the output of step four (character mapping) in the encryption process.

3. **Character Remapping:** The phonetic based encrypted message is mapped in the reverse direction, using ASCII codes or any other mapping technique, into the intermediate language. The result of this step is an intermediate language text message that has no meaning in the intermediate language. At this stage, only humans (and some multilingual intelligent interpreters) who understand both

intermediate and original language can understand the original message. In figure 2, the ET(L1) which is the encrypted text in the original language will be reversely mapped to characters from the selected intermediate language. It should be clear the intermediate language should be the same as in the encryption process. The location mapping between characters should use the inverse of the technique used during the encryption. If the technique uses variable location shifting, the shifting should be done in the opposite direction in order to achieve the correct mapping. The output, T(L2), of this step is a text in the intermediate language that corresponds exactly to the output of step three in the encryption process.

4. **Text to Speech Converter:** Using a text to speech converter of the intermediate language, the generated text in the previous section is converted into speech. The speech created, in this step, might be understood by any human or language interpreter who speaks the original message language. In Figure 2, the TTS(L2) block will receive T(L2) of the previous message and convert it into a speech that corresponds to the acoustic representation of the message S(L2). This representation is meaningless in the intermediate language. However, multilingual humans and speech to text converter of the original language can interpret the message. The output of this step is similar to the output of step 2 in the encryption process.
5. **Speech to Text Converter:** Using a speech to text converter of the original language, this time, the generated speech in the previous step is converted into text. The generated text in this step should be the original text before encryption. Up to this point the decryption process is finished, however, for more accuracy, a verification process should be performed. In figure 2, the S(L2) is processed through STT(L1) resulting in T(L1) which corresponds to the original input to the encryption system. The input that used in both step one and two in the encryption process.
6. **Verification:** To verify the originality of the text, the same private key should be applied to the original text resulting in the variable ciphering key used in the first step of the encryption.

The generic encryption/decryption described above is used for offline text. If the phonetic system is used for online encryption, the protocol should be modified so that the ciphering key used in the second part should be applied directly to the text after performing the phonetic encryption. The ciphering key in this case should be identified in a different manner. The key might depend directly on the private key. Note that, in the online case, message exchange between humans, the human may replace the text-to-speech then speech-to-text conversion steps and perform the operation by a multilingual keyboard. The next section shows a complete example for encryption/decryption process.

4. A Complete Example

In this section a complete example of the proposed protocol is demonstrated. The message to be encrypted will be: “I don't know which direction I should choose; the right one, where nothing is left; or, the left one where nothing is right”. The example will use a fixed location mapping for simplicity. However, it is recommended to use variable location mapping for better attack resistance.

4.1. The Encryption Process

1. The private key is used to create the ciphering key. Once the ciphering key is created, it is saved to be used in the second phase of the encryption. For

simplicity, the private key is the word PRIVATEKEY that is applied to the text and resulted in a ciphering key, which is CIPHERINGKEY, for simplicity.

2. Apply a text to speech conversion, to convert the mentioned phrase above. The result of this step is a sound file that contains vocal version of the phrase.
3. Apply a speech to text conversion that uses the sound file produced from the step 2 and convert it into a text of another language. In this example the speech to text converter will convert the English speech to Arabic text. The output of this step will be “ اي دونت نوهويتش ديركشن اي شود تشوز ذا رايت هوير ناثينج ايز لفت اور ذا لفت هوير ناثينج ايز رايت". This output corresponds vocally to the phrase mentioned above.
4. Apply a mapping technique to map the Arabic text above to an English one. In our case, we use the keyboard direct mapping for simplicity. This will result in the following text, if we use an English keyboard. “hd ^;kj k; I;djq ^hdvmqk hd q;^jq;: ^{2}h vhdj I;dv khedjk^hd: gtj h;v ^{2}h gtj icdv khedk^hd: vhdj” up to this point the first step which is the vocal encryption is done.
5. Apply ciphering by choosing the write N-TEA encryption dictionary and perform one to one encryption of the message. The output of the encryption might be something like:

   “kdlsQURikendQikeQSDFrdklzedikedirif,;vbqwfarjùaeke,viekfrtuDeTdfF”

6. Finally, embed the ciphering key into the message using the private key. It is very important step, so that the receiver can extract the ciphering key to be used in the first step of the decryption. However, the decision is to use the private key directly as a ciphering key.

For simplicity, the final step might be omitted, so that the ciphering key will be embedded in fixed positions that correspond to the numerical order of the private key in the alphabetic order. This assumes that the length of their ciphering key is the same length of the private key.

If the private key is the word PRIVATEKEY and ciphering key is CIPHERRKEY (notice that we added another R to make sure that the ciphering key meets the length of the private key. This process is to be done in the first step and should be respected in the step number five while using the N-TEA for encryption) then for the first letter of the CIPHERRKEY which is C, it will be embedded into the encrypted message in the position number P (or in other words, the position of letter P in the English alphabet that is 16). The same action will be repeated for each letter of the ciphering key.

This process will be restarted at each time for each letter and the embedded letters in the previous iterations are counted as part of the original message. The decoder should do the same process in an opposite way, that is to say, while extracting the ciphering key, the decoder will start from the last key and not from the first one. In our case, the extraction will start with the letter Y and not with the letter P.

4.2. The Decryption Process

The decryption process is exactly the opposite of the encryption process. The received message is encrypted using a ciphering key that is embedded inside it.

1. Extract the ciphering key from the received message and then omit it from the message. As mentioned above, if the private key is PRIVATEKEY, the first step is to search the position number Y (or the position that is indicated by the order of the letter Y in the English alphabet which is 25) then extract the letter that is in that position, this will indicate the last character of the ciphering key. This letter should be deleted from the message before restarting the same process again until the first character of the private key. At this point, we will have the message

exactly as it was produced in the step five of the encryption procedure. It should be as follows:

“kdlsQURikendQikeQSDFrdklzedikedirif,;vbqwfarjùaeke,viekfrtuDeTdfF”
that is the encrypted message using the N-TEA algorithm

2. Apply the ciphering key and perform the first part of the decryption which will result in the message as shown in the step number four of the encryption algorithm which is:“hd ^;kj k; I;djq ^hdvmqk hd q;^jq;: ²h vhdj I;dv khedjk^hd: gtj h;v ²h gtj icdv khedk^hd: vhdj”

3. Apply the keyboard remapping between a English keyboard and an Arabic one to produce the Arabic text that has no meaning in the Arabic language, however, it correspond vocally to the original English message the output of the mapping process will be : “ اي دونت نوهويتش ديركشن اي شود تشوز ذا رايت هوير ناثينج ايز لفت اور ذا
لفت هوير ناثينج ايز رايت"

4. Convert the Arabic text into speech to produce a sound file that corresponds to the original message.

5. Convert the sound file into text again of the original message language to retrieve the original text.

As an optional verification step, the private key should be used on the original message to make sure that the ciphering key is the correct one.

5. Advantages and Challenges

Several challenges exist in this work, especially in two major domains the fist domain is the speech to text conversion when done for meaningless text. The intelligent speech to text converters will not perform correctly and will result in errors in both encryption and decryption processes. However, this challenge makes the task of the intruders harder.

The algorithm is scalable and the process of the vocal encryption might be nested as many times as needed with as many languages as needed. Besides, the second part of encryption is completely independent of the first one and may use any known text encryption algorithm without affecting the first part of the encryption.

The challenge of this algorithm might appear on the mapping process if there are ambiguities of character mapping. A clear example of that is the letters ‘g’, ‘h’, and ‘b’ in the keyboard mapping between the English and Arabic keyboards (in Arabic called 'lam alef' and might be represented in two different ways either by typing the two keys ‘g’ and ‘h’ or just typing the key ‘b’). For each language, separation keys should be identified to mention which of the keystrokes will be used. Direct ASCII mapping could be the solution of such a challenge.

6. Conclusion

This paper proposed a hybrid encryption protocol that merges both acoustics and traditional encryption algorithms. The developed technique uses different languages as intermediate step for encryption. The protocol used multilingual keyboard as an example of mapping between different languages. The protocol is scalable and not limited to only two language mapping. The transitional part of the encryption is independent of the vocal one, which means that any known encryption algorithm might be used on top of the vocal encryption one.

This work opens a venue of research on the phonetic/acoustics based encryption. Further work should address the challenges of multiple interpretations of complex letters in some languages without a need of direct character mapping. The keyboard mapping variable location shifting is another promising venue of research.

References

[1] M. Ogburn, C. Turner and P. Dahal, "Homomorphic Encryption", Procedia Computer Science 20 © Elsevier, **(2013)**, pp. 502 – 509.

[2] P. V. Parmar, S. B. Padhar, S. N. Patel, N. I. Bhatt and R. H. Jhaveri, "Survey of Various Homomorphic Encryption algorithms and Schemes", International Journal of Computer Applications, vol. 91, no. 8, **(2014)**, pp. 26-32.

[3] X. Yi, R. Paulet and E. Bertino, "Homomorphic Encryption and Applications", Springer Briefs in Computer Science, **(2014)**, pp. 27-46.

[4] R. M. Pandav and V. K. Verma, "Data Encryption Using Various Cryptography Techniques: A Recent Survey", IJREAM International Journal for Research in Engineering Application & Management, vol. 1, no. 9, **(2015)**, pp. 1-4.

[5] P. Mishra, "A Comparative Survey on Symmetric Key Encryption Techniques", IJCSE International Journal on Computer Science and Engineering, vol. 4, no. 05, **(2012)**, pp. 877-882.

[6] P.-S. Chung and M.-S. Hwang "A Survey on Attribute-based Encryption Schemes of Access Control in Cloud Environments Cheng-Chi Lee1", International Journal of Network Security, vol. 15, no. 4, **(2013)**, pp. 231-240,

[7] C. Fontaine and F. Galand, "A Survey of Homomorphic Encryption for Nonspecialists", EURASIP Journal of Information Security, vol. 2007, **(2007)**, pp. 1-15.

[8] A. T. Kabakus and R. Kara "Survey of Instant Messaging Applications Encryption Methods", EJOSAT European Journal of Science and Technology, vol. 2, no. 4, **(2015)**, pp. 112-117.

[9] A. M. A. Mansour and M. Fouad, "N-TEA: New Text Encryption Algorithm—Secured Data Exchange", International Journal of Software Engineering and Its Applications (IJSIEA), vol. 8, no. 9, **(2014)**, pp 199-206.

[10] A. M. A. Mansour and M. Fouad, "Bandwidth Optimization for Real-Time Online and Mobile Applications", New Perspectives in Information Systems and Technologies, Volume 2, Series: Advances in Intelligent Systems and Computing, vol. 276, **(2014)**, pp. 281-288.

[11] A. M. A. Mansour and M. Fouad, "Dictionary Based Optimization for Adaptive Compression Techniques", 36th International Convention on Information & Communication Technology Electronics & Microelectronics (MIPRO), **(2013)**; Opatija, Croatia.

[12] A. M. A. Mansour and M. Fouad, "A New Compression Framework: 'A-M' Adaptive Code", 5th International Conf. on Modeling, Simulation and Applied Optimization (ICMSAO), **(2013)**; Hammamet, Tunisia.

[13] B. Wyseur, K. Wouters and B. Preneel, "Lexical Natural Language Steganography Systems with Human Interaction", Proceedings of the 6th European Conference on Information Warfare and Security, **(2007)**, pp. 303-312.

[14] T. Osamu, Y. Akihiro and M. Kyoko, "Secret Sharing Scheme Using Natural Language Text", Journal of the National Institute of Information and Communications Technology, vol. 52, no. 1&2, **(2005)**, pp. 173-183

[15] S. Almuhammadi and A. Al-Shaaby, "A Survey on Recent Approaches Combining Cryptography and Steganography", © CS & IT-CSCP, **(2017)**, pp. 63– 74.

[16] X.-H. Jing, Y. Hao, H.-P. Fei and Z.-J. Li, "Text Encryption Algorithm Based on Natural Language Processing", Fourth International Conference on Multimedia Information Networking and Security (MINES), **(2012)**, pp. 670-673.

[17] R. K. Raphel, H. M. Ilyas and J. R. Panicker, "Multiple Secret Sharing Using Natural Language Letter Based Visual Cryptography Scheme", International Conference on Algorithms and Architectures for Parallel Processing, Algorithms and Architectures for Parallel Processing, **(2015)**, pp. 476-486.

[18] H.-C. Lin, C.-N. Yang, C.-S. Laih and H.-T. Lin, "Natural Language Letter Based Visual Cryptography Scheme", Journal of Visual Communication and Image Representation vol. 24 no. 3, **(2013)**, pp. 318-331

[19] X.-S. Xiao, A. Paradkar, S. Thummalapenta and T. Xie1, "Automated Extraction of Security Policies from Natural-Language Software Documents", SIGSOFT'12/FSE-20, **(2012)**, pp. 1-11.

[20] J. Mason, K. Watkins, J. Eisner and A. Stubblefield, "A Natural Language Approach to Automated Cryptanalysis of Two-time Pads", CCS, **(2006)**; Alexandria, Virginia, USA.