SN Computer Science (2021) 2:46
https://doi.org/10.1007/s42979-020-00438-y

ORIGINAL RESEARCH

Securing Data via Cryptography and Arabic Text Steganography

Malak Alkhudaydi[1] · Adnan Gutub[1]

Received: 14 July 2020 / Accepted: 22 December 2020

Abstract
This research proposes an efficient security system for hiding classified Arabic text-data on limited processors devices benefiting from the combination of both techniques: light-weight cryptography (LWC) and Arabic text steganography. The work assumes to secure the sensitive text-data on devices suffering low resource allocation by layer of LWC encryption to provide acceptable security. In this work, we use a two-layer technique, where we firstly encrypt the Arabic secret text-data examining effectiveness of different LWC algorithms, i.e. AES, DES, and IDEA; then, embedding the encrypted data into diacritics within Arabic text cover media. The work tested the possibility of accepting LWC security of AES, IDEA, and DES encryption assuming their suitable effect on the text stego-cover. The work runs numerous different experimentations to study the preference, where LWC selection of DES algorithm finds the applicable scheme giving acceptable security in efficient manner.

Keywords Arabic text steganography · Information security · Diacritics steganography · Data encryption · Lightweight cryptosystems · AES · DES · IDEA

Introduction

With Internet reaches almost every corner of the planet and more and more of our life aspects are becoming digital. Therefore, expanding our virtual presence, such as email messages, health information, family private pictures, bank information, and credit card information has become essential to transfer and store them in a secure manner [1]. In fact, the security of any data that can be sabotaged is considered a cybercrime as it affects people's life by manipulating classified educational information [2] and e-health information [3]. All data have become digital, making a huge amount of sensitive places vulnerable to security threats and hacking challenges and thereby affecting information on all PCs or handheld smart devices [4]. In this regard, various methods have been provided to protect the information or educate its proper utilization [5] as well as requesting limited capability tools such as cryptography, steganography, and watermarking, to secure the work [6]. In this research, we focus on combining appropriate cryptography and steganography to increase the layers of security dedicated to serve sensitive Arabic text data running on limited capability devices.

Cryptography can be understood as converting the secret clear text to form useless cipher text data using secret key for running the process, i.e. to secure the secret data. We proposed a two-layer Arabic security system applying confusion to the sensitive text data going through the crypto layer, involving encryption security key, then followed by an Arabic text steganography layer to produce the stego file, as overview analogy shown in Fig. 1. The research considered that both crypto and stego layers run on limited computational situations, e.g. handheld smart devices, suffering from restricted memory, and inadequate computational power, as normally forced to struggle for accomplishing the appropriate trade-off between cost and security [4]. Therefore, lightweight cryptography (LWC) is involved as characterized crypto calculation appropriate for restricted asset compelled condition. LWC is used for several crypto examples, such as, therapeutic sensors, RFID labels, and convenient human services gadgets [6]. This work scope used three famous LWC algorithms, namely, AES, DES, and IDEA as recommended to be useful for similar applications [7]. These chosen LWC

✉ Adnan Gutub
aagutub@uqu.edu.sa

Malak Alkhudaydi
malak-alkhdidi@hotmail.com

[1] Computer Engineering Department, Umm Al-Qura University, Makkah, Saudi Arabia

Published online: 19 January 2021

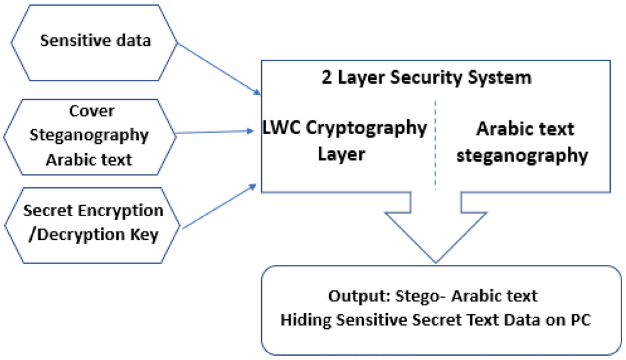

Sensitive data
Cover Steganography Arabic text
Secret Encryption /Decryption Key
2 Layer Security System
LWC Cryptography Layer
Arabic text steganography
Output: Stego- Arabic text
Hiding Sensitive Secret Text Data on PC

Fig. 1 Overview of the two-layer crypto stego security system

techniques are characterized with security, capacity, and reliability, for efficient selection to be used as the applicable preference.

Steganography, as a method of organizing the existence of secret data to be hidden [8], is the second layer of this Arabic security system. The secret text is placed in a seemingly harmless container in such a way that it would be difficult for an outside observer to notice the presence of it, i.e. as built-in secret message. When using computer technology, sound, graphic, and text files can act as stego-cover file containers [9], where if, figuratively speaking, cryptography makes the understandable incomprehensible, then steganography makes the visible as invisible (sometimes in the literal sense of the word). This is achieved by "dissolving" the hidden information among other data of a much larger volume [10].

The main improvement in this security study compared to other previous work that use Arabic text steganography is ensuring that even if the embedded text is discovered, the content remain unknown, as the hidden information has been LWC encrypted. Also, the use of LWC ensures the use of only the minimum amount of computation resources, helping to maintain acceptable limited device capabilities and fulfilling nowadays user experience [4]. Therefore, in this work, we establish efficiency comparison between the three algorithm, AES, DES and IDEA from a capacity, security and reliability perspectives, to determine which can perform better involved with the steganography stage.

Recall the outline of the proposed framework utilizing steganography and cryptography two-stage system, illustrated in Fig. 1; it is different than previous works although benefited from all. Our work is different than Arabic stego presentations of hiding within Kashida [11], diacritics [10] and using multipoint letters [12] as well as lightweight crypto [7] were all are single layer security strategies. It is also different than the images security using two-layers [13] and three-layers [14] in our hiding within texts and selecting dedication to limited capability devices utilizing LWC diacritics combination.

The flow of the paper is as follows. “Related Background” covers the background related to LWC and Arabic text steganography building the different blocks making this integration work. “Proposed Approach” describes the proposed combination security approach defining both hiding and retrieving security processes in depth. “Implementation Performance” covers the implementation and performance experimentations remarks presenting the capacity, security, and reliability efficiency measurements. Then, “Conclusion” concludes the paper.

Related Background

This research of securing data via lightweight cryptography and Arabic text steganography is developed from many previous stego crypto researches. In other words, we are trying to select benefits from both cryptography and security intended to run over limited capability devices, i.e. providing acceptable security, similar in principle to image security research of [13, 14]. This section will go over related background attempts of lightweight cryptography as well as Arabic text steganography to partially select from and combine and thereby evolving our new contribution.

Lightweight Cryptography Algorithms

Lightweight cryptography (LWC) focuses on symmetric crypto-algorithms intended for limited capability devices [15]. The chosen three LWC algorithms for this study are AES, DES, IDEA. The AES (Advanced Encryption Standard), is the crypto method commonly adopted worldwide using a 128-bit fixed block size to produce its encrypted output. It can use up to 14 transformation rounds using 256-bit key, with each round consisting of several processing steps. The DES (Data Encryption Standard) is one of the fundamental building blocks used in today’s life applications cryptography. It needs an encryption key with 56-bit size, less than AES, as still used to provide minimum-security level [15]. The IDEA (International Data Encryption Algorithm) is closely formatted to AES but uses 128-bit key arranged for 64-bit fixed block size running 8 transformation rounds. It is reported that IDEA level of security is below AES, and above DES making the choice to select our study on three used applicable crypto schemes [7].

In this work, we establish a comparison platform to efficiently select between the three crypto-algorithms from capacity, security, and reliability perspectives. The work is specific towards Arabic text hiding to determine which can perform better combining LWC with Arabic steganography. It is to be noted about LWC that lightweight idea is not less secure; however, it is giving crypto security less computation need allowing expending for momentary treatment fulfilling the time condition of crypto protection, i.e. not decreasing the security or protection for today's applicability [15].

Arabic Text Features

Arabic is the language used by 1.7 billion Muslims around the world [16]. It is the sixth most spoken language having standard 28 letters, written from right to left, opposite to English. Interestingly, Arabic letters have different shapes, depending on their position in the word. In addition, the Arabic language is characterized by series of pointed letters (one, two or three dots above or below certain letters), as shown in Fig. 2. All Arabic words contain several letters involving optional vowels written in specific way. These optional vowels called "Harakaat" are our useful location for hiding data. These vowels "Harakaat" are considered as text-characters with diacritical marks but does not affect the wording much.

There are 8 Arabic diacritics, i.e. Fathah, Kasrah, Dammah, Fathatan, Dammatan, Kasratan, Sukun and Shaddah, as shown in Fig. 3. Those diacritics or Harakat have values at the sound, meaning and grammar levels in Arabic language. Please note that Arab diacritics are available for current and modern Arabic descriptions [17].

Arabic Text Steganography

Many textual steganography methods are linked to the English language as standard scientific medium, but few current research consider Arabic textual steganography techniques in Arabic and identify limitations in terms of security, capacity and reliability as noted in researches of [12, 18–20].

The methods of steganography in the Arabic text can be classified into the several types. For example, Mohamed [21] classified Arabic steganography according to shifting points. Gutub and Al-Nazer [11] and Odeh et al. [12], discussed the Kashida and Arabic diacritics-based (Harakat) steganography similar in principle to ours. Bensaad and Yagoubi [22] worked on Kashida-based steganography based on Al-Haidari et al. [23]. Later, Al-Azawi and Fadhil [24], presented Unicode Arabic stego method. Shirali-Shahreza [25] and Por et al. [26] as well as Mohamed [21] formed linguistics stego schemes similar to Desoky [27] and Alabish et al. [28], but with furthermore interesting features. For instance, Mohamed [21] discussed a method for the Arabic steganography using isolated letters to form words of data hiding. This technique provided low bandwidth relatively safe algorithm compared to other researched techniques.

Por et al. [26] projected previous models to encode data by injecting unusual Unicode fonts in spaces between phrases. Therefore, hiding data required modifications to the content which cannot be easily figured out. Similarly, Kadhem and Ali [19] discussed a tool that takes all cover texts (characters, diacritical marks, spaces) and uses them as covering means, which increases the capacity influence with large amount of showing content.

Al-Nofaie et al. [18] suggested a technique to hide secret bits within spaces between Arabic words. Although their process rises the risks of cover-up security, it requires clear alteration within cover showing increase in its size. Lately, Malik et al. [20] introduced a scheme utilizing color-coding table. Their model had a clear problem in changing the colors that attracts attention.

Our system is proposed to utilize diacritics (vowel signs) by showing or omitting them to hide bits. The method usefulness can be considered via interesting capacity utilizing all possible diacritic used to hide bits. In fact, even excluded or omitted diacritics will be hiding bits benefitting the best possible performance. The security and reliability measures are proving the work applicability in an interesting manner as will be discussed later.

Proposed Approach

The proposed idea comes from utilizing the actual display of Arabic diacritics similar in principle to the 2010 work of Gutub et al. [10]. In general, for most Arabic e-text sources, if a diacritic marker code is found, the corresponding line image is displayed without changing its cursor position. This unbiased visualization opens the possibility of involving some diacritic signs without much notice. We will utilize computer programs that recognize the existence of such diacritics to recognize and interpret them to our objective.

Pointed letters	un-pointun letters
ش ز ذ خ ج ث ت ب ي ن ق ف غ ظ ض ة	ص س ر د ح أ و ه م ل ك ع ط

Fig. 2 Pointed and un-pointed Arabic letters

Fig. 3 Arabic text Diacritics

Fathah ◌َ	Dammah ◌ُ	Kasrah ◌ِ	Sukun ◌ْ
Tanween Fathah ◌ً	Tanween Dammah ◌ٌ	Tanween Kasrah◌ٍ	Shaddah◌ّ

A steganographic system, or stegosystem, is a set of means and methods that are used to form a hidden channel for transmitting information. When building a stegosystem, the following provisions should be taken into account:

1. The adversary has a complete understanding of the steganographic system and the details of its implementation. The only information that remains unknown to the potential adversary is the key, with the help of which only its holder can establish the fact of presence and content of the hidden message.
2. If the adversary somehow finds out about the existence of a hidden message, this should not allow him to benefit from messages in other data as long as the key is kept secret.
3. A potential adversary should be deprived of any technical or other advantages in recognizing or positively disclosing the content of secret messages.

Therefore, stego-systems normally follow the below requirements [17]:

1. The properties of the container must be modified so that the change could not be detected by visual inspection. This requirement determines the quality of concealment of the message being introduced: to ensure unhindered passage of the stego message over the communication channel.
2. Stego message should be resistant to distortion, including malicious. In the process of transferring an image (sound or another container) it can undergo various transformations: decrease or increase, convert to another format, etc.
3. To maintain the integrity of the embedded message, we may use error correction code.
4. To increase reliability, the embedded message can be duplicated.

Our proposed system utilizes both cryptography and steganography to benefit from both as detailed in Fig. 4. Cryptography and steganography are both exploited as separate layers to give the best practical security with independent measurable security, capacity, reliability measures and improvement adjustments. In the following subsections, we will explain our algorithm, its architecture, components, and data. Also, explain the main processes of the algorithm, hiding process and retrieving process.

Algorithm Architecture

The two-layers system can be observed as a process flow graph clarifying the storing point of view (Fig. 5) as well as the retrieving point of view (Fig. 6). The cryptography layer is using the well-known standard AES, DES and IDEA crypto algorithm, as commonly selected efficient [7]. Note that the encryption secret key used in the storing flow graph is also needed in the decryption process, i.e. when retrieving the data.

The sender of the hidden sensitive secret text uses the storing process as in Fig. 5. The recipient can retrieve the

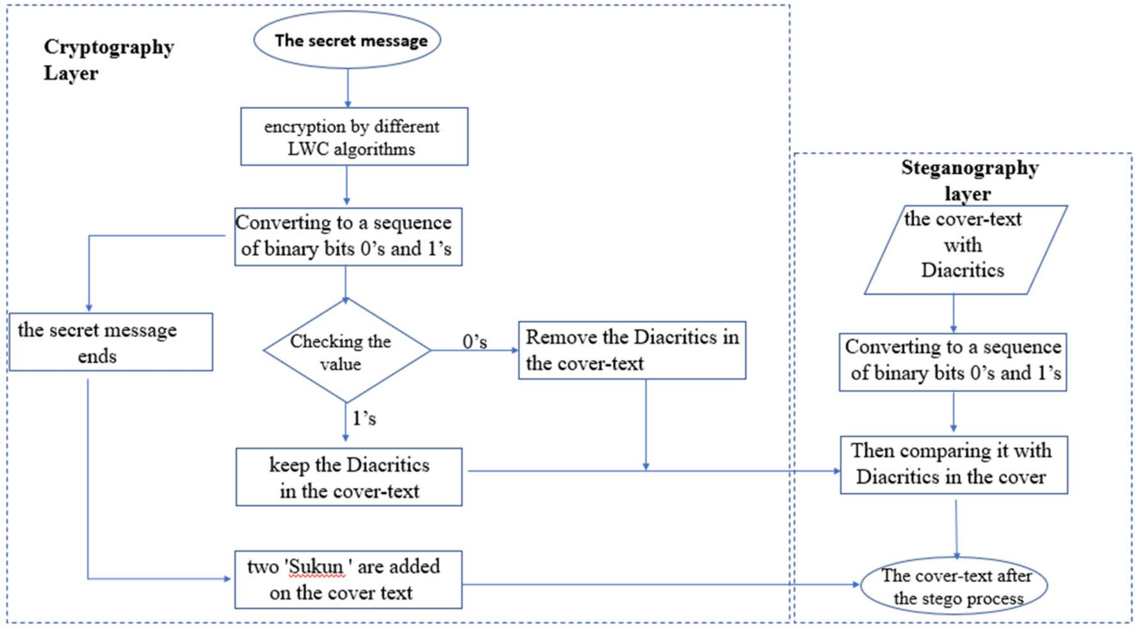

Cryptography Layer
The secret message
encryption by different LWC algorithms
Converting to a sequence of binary bits 0's and 1's
the secret message ends
Checking the value
0's
Remove the Diacritics in the cover-text
1's
keep the Diacritics in the cover-text
two 'Sukun ' are added on the cover text
Steganography layer
the cover-text with Diacritics
Converting to a sequence of binary bits 0's and 1's
Then comparing it with Diacritics in the cover
The cover-text after the stego process

Fig. 4 Block diagram of the proposed crypto stego security system

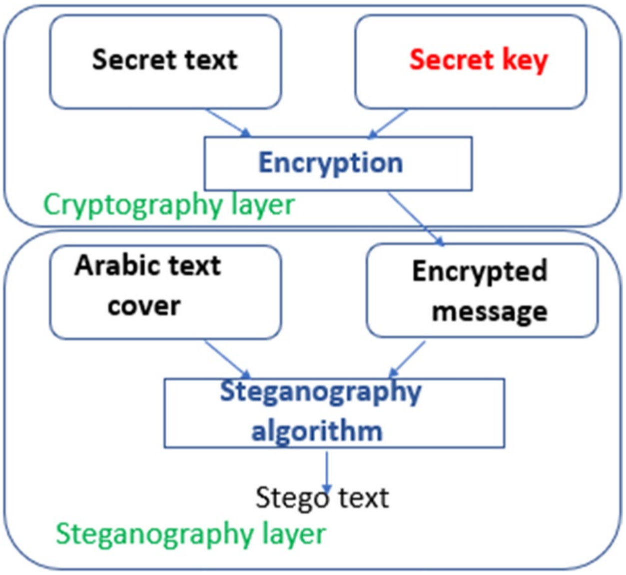


Fig. 5 Storing sensitive secret text data

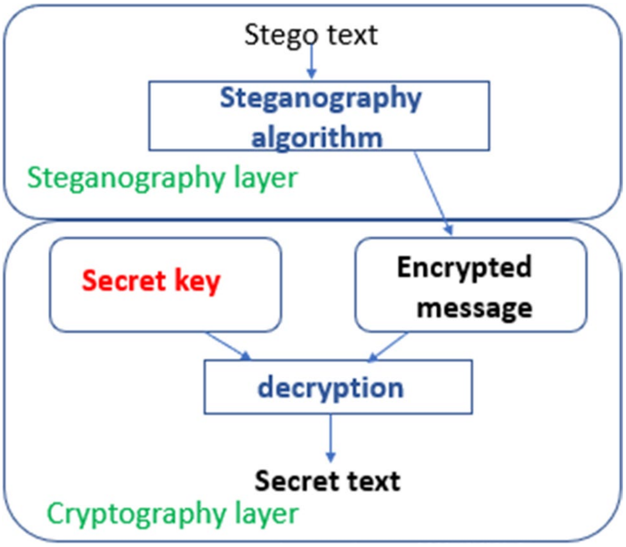


Fig. 6 Retrieving back secret text data

encrypted secret text from the steganography Arabic text as the retrieving process shown in Fig. 6. The hiding secret text follows symmetric encryption as standard method. Then, our proposed focus of hiding in Diacritics, (Fathah, Kasrah, Dammah, Fathatan, Dammatan, Kasratan, Sukun and Shaddah) is to take place via steganography. This process will result a stego-text, which can be distributed publicly aiming unsuspicious actions.

The retrieving process uses the same stego-text, which is retrieved; it will retrieve the encrypted secret text. The extracted secret text needs decryption to present the final output from this same symmetric algorithm applied. Note that this crypto algorithm is affecting the study analysis of the results on the capacity of hiding places, i.e. compared between different LWC encryption algorithms. In the next two sub-sections, we will explain both processes, which represent the main functions of the algorithm.

To summarize our focus, we will present diacritics as cover to hide the information. Our proposed diacritics algorithm is to hide secret binary data into Arabic text. The Arabic letters and diacritics are then converted to binary bits and the secret text is stored. It identifies the cover text that will be used in the hiding process. The user has the option to select text file for storing data based on his device, or to enter the cover Arabic text manually. The beneficial assumption of this study is that all Arabic text cover should involve diacritics. This process is important to ensure that the positions of the secret bits are placed within the applicable range of the cover text.

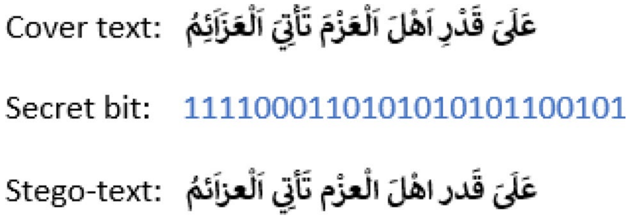
Hiding Process

This process involves allowing selection of the input file, which represents the secret message. Then, encryption algorithm and identification of cover text used in hiding message is assigned. The research security and capacity calculations of bit hiding is going to be our focus. The output of this process will be a stego text, which will be used by the recipient to retrieve the secret message again.

To detail the proposed algorithm, it starts by converting the cover-text to binary bits. Then, it is storing the bits as an array, where we consider every 16-bits of which to represent an Arabic letter or diacritics which are placed visually together. In addition, 8-bits are added representing the all non-Arabic characters, to complete the programming proper investigation. Recall that the secret message is encrypted by the intended LWC algorithms. Our program reads the first bit of the secret message and then compares it with the first diacritics in the cover-text. If, for example, the first bit to be hidden was a 'one', this first diacritic, say 'Fathah', will remain; otherwise, the 'Fathah' will be removed. This process will repeat itself until all secret bits in the secret message are considered. When the secret message ends, two'Sukun' are added on the cover-text, which is not found in Arabic writing, used as an indication of ending the hiding process. The hiding process can be formalized as pseudo code points below:

1. Conduct secret message encryption by different LWC algorithms.
2. Convert the encrypted message to sequence of binary bits 0's and 1's.
3. Insert cover-text with full Diacritics and convert it to binary bits.
4. Store secret bits as an array.

5. Check value of secret bits and relate them in sequence to Diacritics in the cover.
    a. if secret is ‘1’ keep the ‘Diacritic’ as is.
    b. if secret is ‘0’ remove Diacritic.
6. Repeat step 5 until the secret message is fully embedded.



Cover text: عَلَىَ قَدْرِ اَهْلَ اَلْعَزْمَ تَأْتِي اَلْعَزَاَئِمُ

Secret bit: 1111000110101010101100101

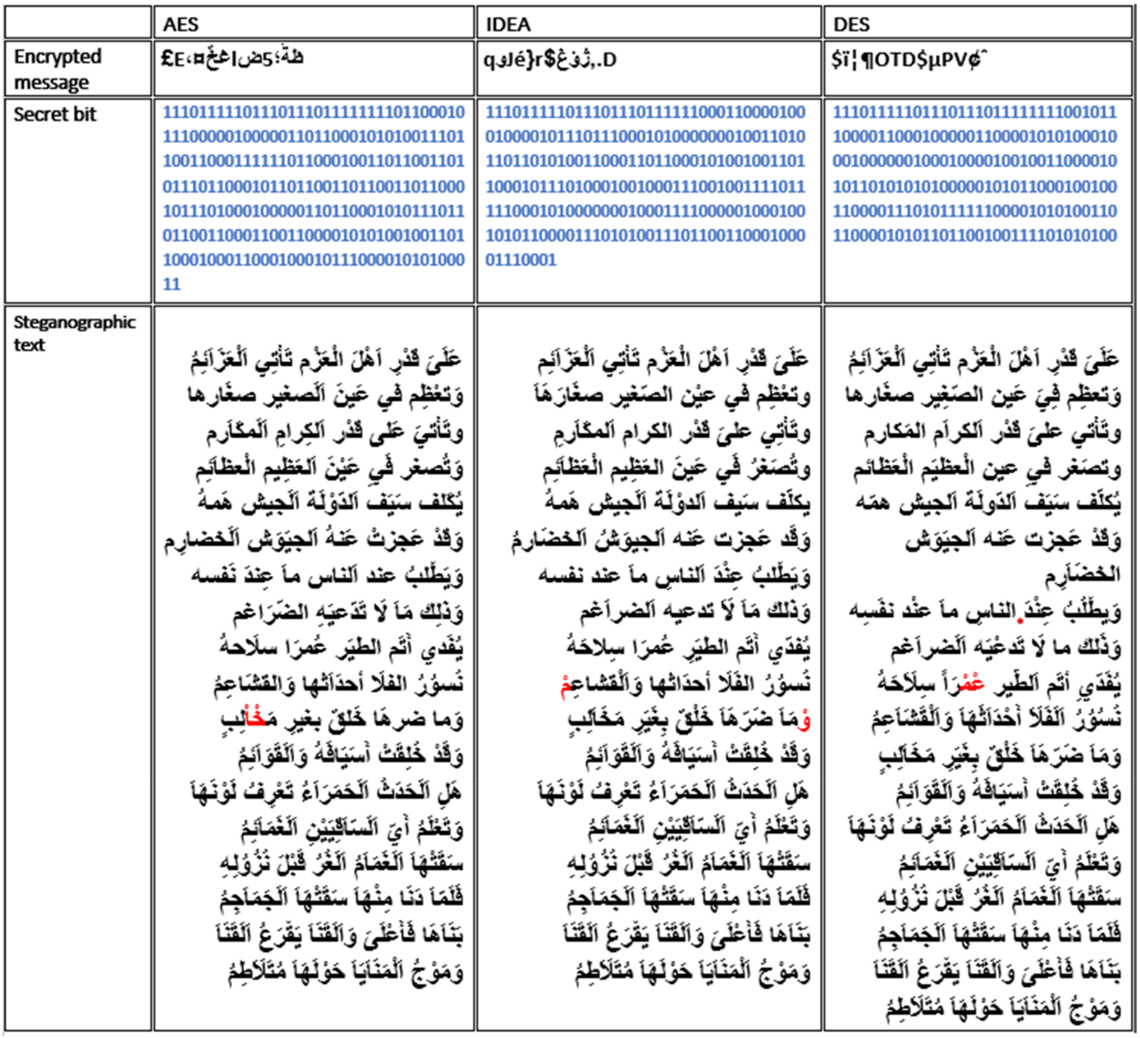
Stego-text: عَلَىَ قَدر اهْلَ الْعزْم تَأْتِي اَلْعزائمُ

Fig. 7 Example of the proposed technique

7. Indicate hiding end by inserting two ‘Sukun’ Diacritics within the cover-text.
8. Show both texts before and after the hiding process.
9. End.

To hide some data in a cover text using the proposed technique, we first have to be sure that all possible diacritics are present in the cover text, then we sequentially match every diacritic to a bit from the secret bits. After that, we apply the hiding process as example shown in Fig. 7.

The secret message used in this example (Fig. 7) is normal name “Malak”. The algorithm encrypted the secret message by LWC algorithm then converted the secret message to binary bit as detailed in Fig. 8 assuming the different LWC algorithms.



	AES	IDEA	DES
Encrypted message	£E‹¤اعغ|ض5؛طة	qوJé}r$غ,ژوغ,.D	$ï¦¶OTD$µPV¢ˆ
Secret bit	1110111110111011101111111101100010 1110000010000011011000101010011101 1001100011111101100010011011001101 0111011000101101100110110011011000 1011101000100000110110001010111011 0110011000110011000010101001001101 1000100011000100010111000010101000 11	1110111110111011101111111000110000100 0100001011101110001010000000010011010 1101101010011000110110001010010001101 1000101110100010010001110010011111011 1110001010000000100011110000010001000 1010110000111010100111011001100010000 01110001	1110111110111011101111111111001011 1000011000100000110000101010001010 0010000000100010000100100110000100 1011010101010000010101100010010010 1100001110101111110000101010011010 1100001010110110010011110101010100
Steganographic text	عَلَىَ قَدْرِ اَهْلَ الْعَزْم تَأْتِي اَلْعَزَاَئِمُ وَتعْظِمِ فِي عَيْنَ اَلصغير صغَارها وتَأْتيَ عَلى قَدْر اَلكِرامِ اَلمكَاَرم وَتُصغر فَيِ عَيْنَ اَلعَظِيم الْعظاَئِم يُكلف سَيَف اَلدوْلَةَ اَلجيش هَمهُ وَقَدْ عَجزتْ عَنهُ اَلجيَوش اَلخضارِم وَيَطْلبُ عند اَلناس ماَ عِندَ نَفسه وَذلِك مَاَ لَا تَدَعيَهِ الضَرَاغم يُفَدي أتَم الطيَر عُمرَا سلَاحهُ نُسوُرُ الفلَا أحدَاَثها وَالقشَاعِمُ وَما ضرهَا خَلقٌ بغيرِ مَخاْلِبِ وَقَدْ خَلِقَتْ أسَيَافُهُ وَاَلقَوَاَئِمُ هَلِ اَلحَدَثُ اَلحَمَرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أيَ اَلسَاقِيَيْنِ اَلغَمَاَئِمُ سَقَتْهَاَ اَلغَمَاَمُ اَلغُرُ قَبْلَ نُزُوُلِهِ فَلَمَاَ دَنَا مِنْهَاَ سَقَتْهَاَ اَلجَمَاَجِمُ بَنَاَهَا فَأعْلَىَ وَاَلقَنَاَ يَقْرَعُ اَلقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلَاَطِمُ	عَلَىَ قَدْرِ اَهْلَ الْعَزْم تَأْتِي اَلْعَزَاَئِم وتعْظِمِ فِي عيْن الصَغير صغَارَهَاَ وتَأْتِي علىَ قَدْر الكرام اَلمكَاَرِمِ وتُصَغرُ فَيِ عَينَ العَظِيم الْعَظاَئِم يكلَف سَيف اَلدوْلَةَ اَلجيش هَمهُ وَقَد عَجزت عَنه اَلجيوَشُ اَلخضَارمُ وَيَطْلبُ عِنْدَ اَلناسِ ماَ عند نفسه وَذلك مَاَ لَاَ تدعيه اَلضراَغم يُفدَي أتَم الطيَرِ عُمرَا سِلاحَهُ نُسوُرُ الفَلَا أحدَاثها وَاَلْقشاعِمْ وْمَاَ ضَرَهَاَ خَلْقٌ بِغَيَرِ مَخَاَلِبِ وَقَدْ خَلِقَتْ أسَيَافُهُ وَاَلقَوَاَئِمُ هَلِ اَلحَدَثُ اَلحَمَرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أيَ اَلسَاقِيَيْنِ اَلغَمَاَئِمُ سَقَتْهَاَ اَلغَمَاَمُ اَلغُرُ قَبْلَ نُزُوُلِهِ فَلَمَاَ دَنَا مِنْهَاَ سَقَتْهَاَ اَلجَمَاَجِمُ بَنَاَهَا فَأعْلَىَ وَاَلقَنَاَ يَقْرَعُ اَلقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلَاَطِمُ	عَلَىَ قَدْرِ اَهْلَ الْعَزْم تَأْتِي اَلْعَزَاَئِمُ وَتعظِم فِيَ عَين الصَغِير صغَارها وتَأْتي علىَ قَدْر اَلكرَاَم المَكارم وتصَغر فِي عين الْعظيَم الْعَظائم يُكلَف سَيَف اَلدَولَةَ اَلجيش همَه وَقَدْ عَجزت عَنه اَلجيَوش الخضَاَرِم وَيطْلبُ عِنْدَ.الناسِ ماَ عنْد نفَسِه وَذْلك ما لَا تَدعْيَه اَلضراَغم يُفَدَيِ أتَم اَلطَير عَمْرَاَ سِلَاَحَهُ نُسُوُرُ اَلفَلَاَ أحْدَاَثُهَاَ وَاَلْقَشَاَعِمُ وَمَاَ ضَرَهَاَ خَلْقٌ بِغَيَرِ مَخَاَلِبِ وَقَدْ خَلِقَتْ أسَيَافُهُ وَاَلقَوَاَئِمُ هَلِ اَلحَدَثُ اَلحَمَرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أيَ اَلسَاقِيَيْنِ اَلغَمَاَئِمُ سَقَتْهَاَ اَلغَمَاَمُ اَلغُرُ قَبْلَ نُزُوُلِهِ فَلَمَاَ دَنَا مِنْهَاَ سَقَتْهَاَ اَلجَمَاَجِمُ بَنَاَهَا فَأعْلَىَ وَاَلقَنَاَ يَقْرَعُ اَلقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلَاَطِمُ

Fig. 8 Example of the data hiding and cryptography (LWC)

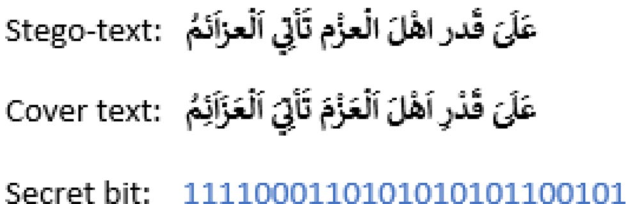
Extracting Process

The extracting process is the reverse of the bit hiding process. It needs the stego-text and the secret message decryption algorithm to be applied. Our software platform is programed to show the verification data, i.e. by pressing the button “show data”, the program will operate retrieving back the secret sensitive text data. It shows the binary bits of the encrypted hidden text within the diacritics. Then, the software converts the binary bits to its original text. Notes that the platform result shows the encrypted text that needs to be decoded with the intended LWC algorithm requiring the secret key as input to the reverse crypto layer. It decrypts the cipher text generating back the secret sensitive data message, as example shown in Fig. 9. In general, to extract the data from the cover text using the proposed technique, we spot the unavailable diacritics as hiding bits of zeroes while others are simply ones. The extracting process can be formalized as pseudo code points below:

```
1. Read Stego-text holding the LWC encrypted secret data.
2. Scan the Arabic characters sequentially noting Diacritics.
   a. if Diacritis is found, store value ‘1’.
   b. if Diacritics is missing, store value ‘0’.
3. Repeat step 2 until two ‘SUKUN’ is detected.
4. Form the stored bits as secret cipher-stream.
5. Conduct secret message decryption by selected LWC algorithm.
6. Show the decrypted secret message.
7. End.
```

Implementation Performance

The two-layer security system for hiding sensitive text data, on personal computers, is implemented on a visual studio-programming platform. The algorithm is programmed by c# language, using UTF-8 Encoding due to its convenience fully supporting Arabic text. We used c# programming language due to its flexibility, wideness spread, and easy to learn. Our objective, since we are in the early stage of this research, is allow any research programmer to simply redesign the system to verify and improve our work. The implementation aim is to test the proposed methodology for hiding information within diacritics and conduct the text encryption by different LWC algorithms, as shown via the 2-layer platform in Fig. 10. The research fixed the cover-text for fair experimentations to the poem shown in Fig. 11.


Stego-text: عَلَى قُدر اهْلَ ٱلعزْم تَأْتِي ٱلعزائمُ

Cover text: عَلَى قُدْرِ أَهْلَ ٱلعَزْمِ تَأْتِي ٱلعَزَائِمُ

Secret bit: 1111000110101010101100101

Fig. 9 Example of the extracting process

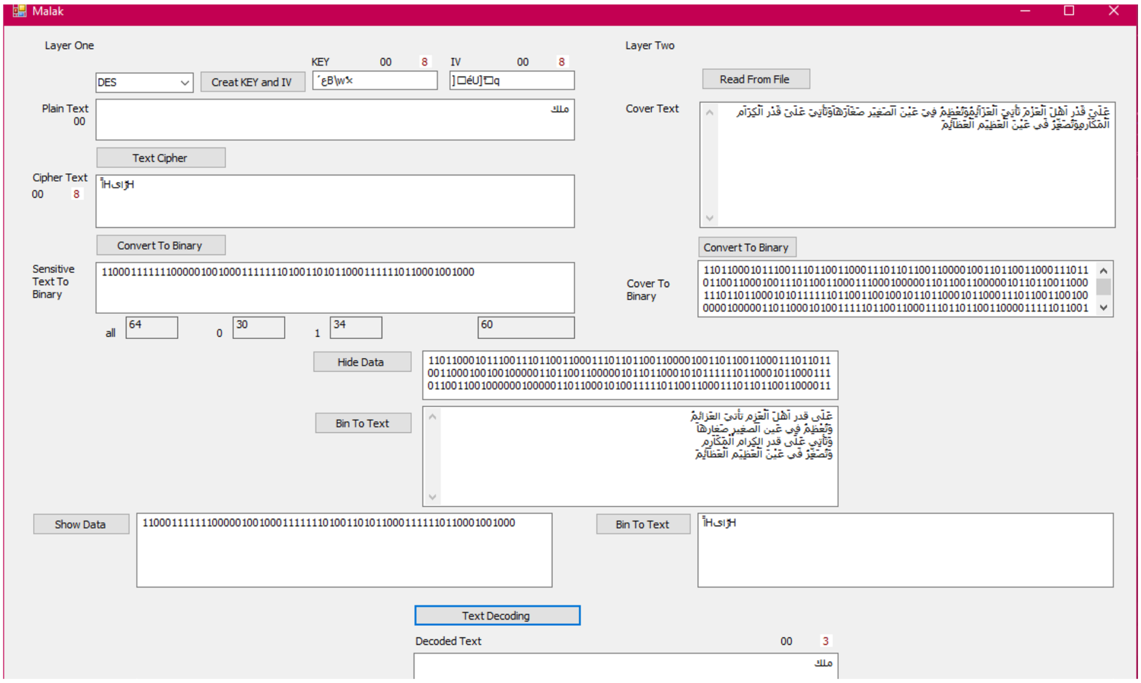
Our testing implementation work used 100 secret messages randomly generated for different lengths. In other words, we selected the secret data as random 3-letter words generated for 100-times and tested among the three LWC algorithms on the same cover-text generating the remarkable results affecting capacity as well as security. Figure 12 shows the three letters secret data (say names) generated randomly for the 100 testing implementations run utilizing all LWC algorithms. Similarly, these 100 testing is further run for other random secret data names assuming Arabic words of 4,5,…14,15,16-letters, all randomly generated and then encrypted by LWC: AES, DES and IDEA showing examination outcomes.

The main improvement in this security study compared to other previous work is that the use of Arabic text steganography is ensuring that even if the embedded text is discovered, the content remain unknown, as the hidden information has been LWC encrypted. Also, the use of LWC ensure to use only the minimum amount of computation resources, helping to maintain acceptable limited device capabilities and fulfilling nowadays user experience [4]. Therefore, in this work, we establish efficiency comparison between the three algorithm, AES, DES and IDEA from a capacity, security and reliability perspectives, to determine which can perform better involved with the steganography stage.

Recall the outline of the proposed framework utilizing steganography and cryptography two-stage system, illustrated in Fig. 1, is different than previous works although benefited from all. Our work is different than stego presentations of hiding within Kashida [11], diacritics [10] and using multipoint letters [12] as well as lightweight crypto [7] were all are single layer security strategies. It is also different than the images two-layer security [13] and three-layers security [14] in our serving texts and selecting dedication to limited capability devices utilizing LWC diacritics combination as elaborated briefly in “Related Background” covering the related background of LWC and Arabic text steganography.

Capacity Testing

Capacity means that the maximum number of bits that can be hidden into the specified cover text. We use the number of zeroes in the sensitive text when converted to binary bits to be our indicator. If the number of zeroes is large, all the diacritics are removed meaning that capacity utilized increased.



DES

KEY 00 8 ´εB\w⅟x

IV 00 8]□éU]□q

Plain Text 00

ملك

Cipher Text 00 8

ÎHىاHز

Sensitive Text To Binary

```
1100011111110000010010001111111010011010110001111110110001001000
```

all 64 0 30 1 34 60

Cover Text

عَلَىَ قَدْرِ اَهْلَ اَلْعَزْمِ تَأْتِيَ اَلْعَزَائِمُوَتَعْظِمُ فِيَ عَيْنَ اَلصَّغِيرِ صَغَارُهَاوَتَأْتِيَ عَلَىَ قَدْرِ اَلْكِرَامِ
اَلْمَكَارِمِوَتَصْغُرُ فِى عَيْنَ اَلْعَظِيمِ اَلْعَظَائِمُ

Cover To Binary

```
1101100010111001110110011000111011011001100001001101100110001110 11
0110011000100111011001100011100010000011011001100000101101100 11000
1110110110001010111111011001100100101101100010110001110110011 00100
0000100000110110001010011111011001100011101101100110000111110 11001
```

```
110110001011100111011001100011101101100110000100110110011000111011011
001100010010010000011011001100000101101100010101111110110001011000111
011001100100000010000011011000101001111101100110001110110110011000011
```

عَلَى قدر آهْلَ آلْعَزم تأتيَ العَزائمُ
وَتَعْظِمُ فِي عَين آلصغير صَغارهَا
وَتَأتِي عَلَى قدر الكِرام آلمَكارم
وَتَصْغُرُ فِى عَيْنَ آلعَظِيم العَظائمُ

```
1100011111110000010010001111111010011010110001111110110001001000
```

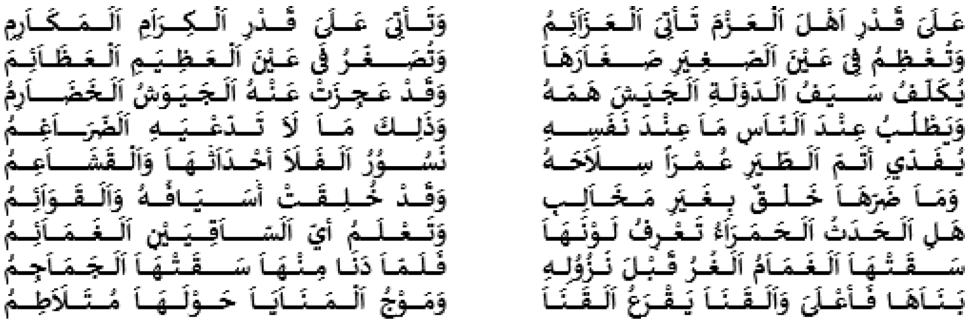
ÎHىاHز

Decoded Text 00 3

ملك

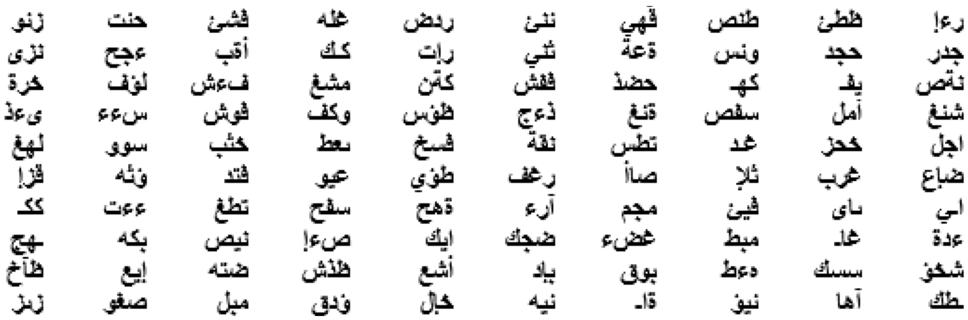
Fig. 10 Proposed security system interface

Fig. 11 Testing cover text used



عَلَىَ قَدْرِ اَهْلَ اَلْعَزْمَ تَأْتِيَ اَلْعَزَائِمُ   وَتَأْتِيَ عَلَىَ قَدْرِ اَلْكِرَامِ اَلْمَكَارِمِ
وَتُعْظِمُ فِيَ عَيْنَ اَلصَّغِيرِ صَغَارُهَا   وَتُصَغَّرُ فِي عَيْنَ اَلْعَظِيمِ اَلْعَظَائِمُ
يُكَلِّفُ سَيْفُ اَلدَّوْلَةِ اَلْجَيْشَ هَمَّهُ   وَقَدْ عَجِزَتْ عَنْهُ اَلْجُيُوشُ اَلْخَضَارِمُ
وَيَطْلُبُ عِنْدَ اَلنَّاسِ مَا عِنْدَ نَفْسِهِ   وَذَلِكَ مَا لَا تَدَّعِيهِ اَلضَّرَاغِمُ
يُفَدِّي أَتَمُّ اَلطَّيْرِ عُمْراً سِلَاحَهُ   نُسُورُ اَلْفَلَا أَحْدَاثُهَا وَاَلْقَشَاعِمُ
وَمَا ضَرَّهَا خَلْقٌ بِغَيْرِ مَخَالِبِ   وَقَدْ خُلِقَتْ أَسْيَافُهُ وَاَلْقَوَائِمُ
هَلِ اَلْحَدَثُ اَلْحَمْرَاءُ تَعْرِفُ لَوْنَهَا   وَتَعْلَمُ أَيَّ اَلسَّاقِيَيْنِ اَلْغَمَائِمُ
سَقَتْهَا اَلْغَمَامُ اَلْغُرُّ قَبْلَ نُزُولِهِ   فَلَمَّا دَنَا مِنْهَا سَقَتْهَا اَلْجَمَاجِمُ
بَنَاهَا فَأَعْلَى وَاَلْقَنَا يَقْرَعُ اَلْقَنَا   وَمَوْجُ اَلْمَنَايَا حَوْلَهَا مُتَلَاطِمُ

Fig. 12 Three letters secret data for the 100 random testing implementations



رءإ	ظطئ	طنص	قهي	نتئ	ربض	غله	قشئ	حنت	زنو
جدر	حجد	ونس	ةعة	ثني	رات	كك	أقب	ءجح	تزى
نقص	يف	كهـ	حضذ	فقش	كتن	مشغ	فءش	لوف	خرة
شنغ	أمل	سقص	قنغ	ذءج	ظوس	وكف	فوش	سءء	ىءذ
اجل	حخز	غد	تطس	نقة	فسخ	نعط	خثب	سوو	لهغ
ضباع	غرب	ثلإ	صاأ	رغف	طوي	عيو	فتد	ونه	قزإ
اسي	ساى	فيئ	مجم	آرء	ةهج	سفح	تطغ	ءءت	كك
ءدة	غا	مبط	غضء	ضجك	ايك	صءإ	نيص	بكه	هج
شخز	سسك	دءط	بوق	ياد	أشع	ظذش	ضته	ايع	ظآخ
طك	آها	نيز	قا	نيه	خال	وذق	مبل	صغو	زرز

The issue comes in picture since the names size change differently after encryption causing this study to reveal unusual results. So the capacity calculation is based on the number of diacritics removed from the cover text. Recall that the Arabic character and diacritics represents 2-bytes in binary code, which are not to be mixed. Our work 100 secret messages of the different secrets lengths (ranging from 3-letters to 16-letters) are all encrypted by AES, DES and IDEA to compare the overall capacity, i.e. of the hidden sensitive text based steganography in the stego layer, to show the best encrypted text of three algorithms in terms of capacity. The capacity assumed Arabic character and diacritics are represented by 16-bits, then divided by 8 to convert them into bytes, calculated as:

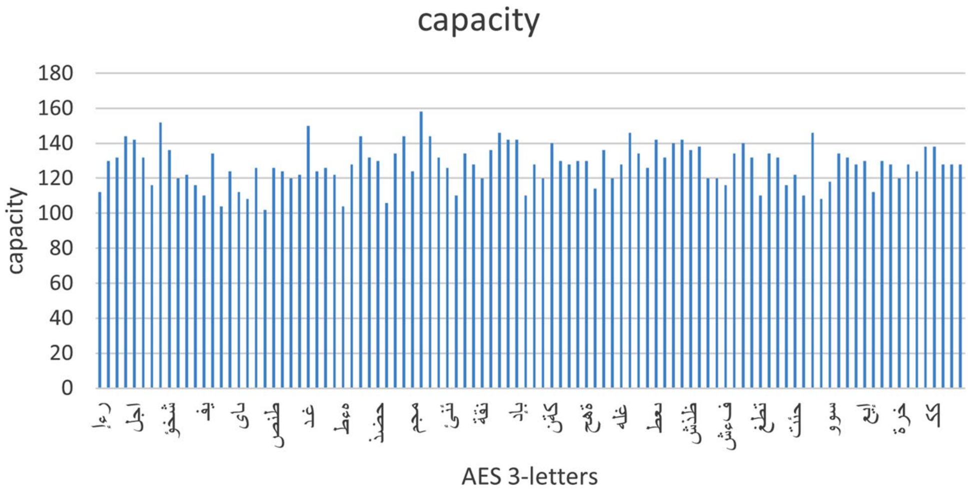


Fig. 13 Capacity for three letters encrypted by AES

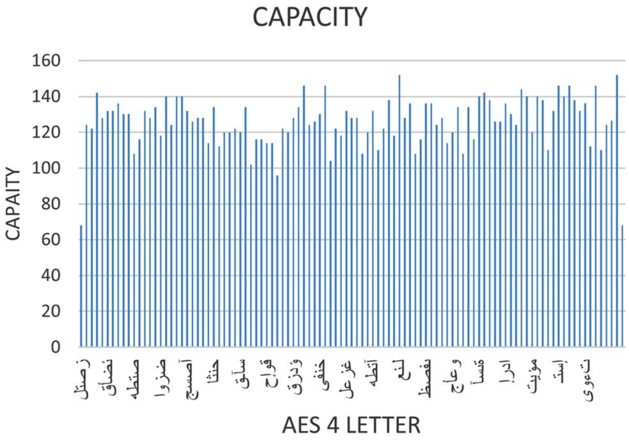


Fig. 14 Capacity for four letters encrypted by AES

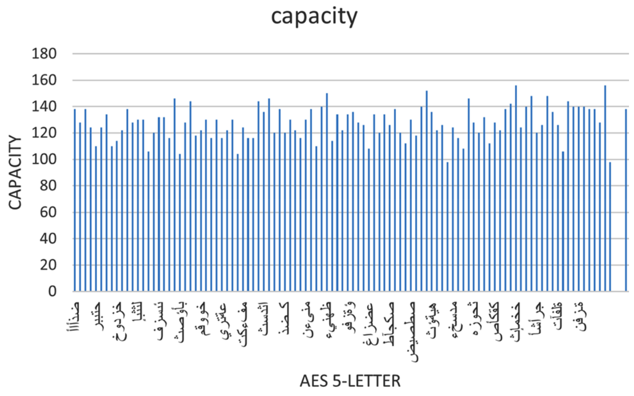


Fig. 15 Capacity for five letters encrypted by AES

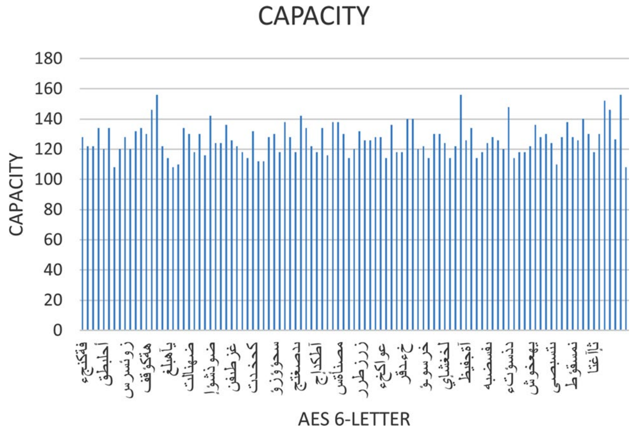


Fig. 16 Capacity for six letters encrypted by AES

$$\text{Capacity} = (\text{number of zeros} \times 16)/8$$

For example, assume that three characters of secret data were selected (Fig. 12) and AES algorithm is used for encryption, the number of characters determined for the secret text for the 100 random names will be generated as length calculation below:

$$\text{The number of zero bits} \\ = 64;\ \text{the capacity} = (64 \times 16)/8 = 128\ \text{bytes}$$

The work run the testing on all randomly generated 100 names of different sizes, i.e. 3-letters to 16-letters, all experimented via the three LWC algorithms: AES, DES and IDEA. Due to the limited space of presentation, we present the AES study to be used as example where all the rest of crypto methods are run the same way. Consider the following AES encrypted capacity figures, i.e. Figure 13 indicating capacity for 3 letters encrypted by AES, Fig. 14 showing capacity for 4 letters, Fig. 15 for 5 letters, Fig. 16 for 6 letters, Fig. 17 for 7 letters, Fig. 18 for 8 letters, Fig. 19 for 9 letters, Fig. 20 for 10 letters, Fig. 21 for 11 letters, Fig. 22 for 12 letters, Fig. 23 for 13 letters, Fig. 24 for 14 letters, Fig. 25 for 15 letters, and Fig. 26 for 16 letters, all representing AES experimentations of

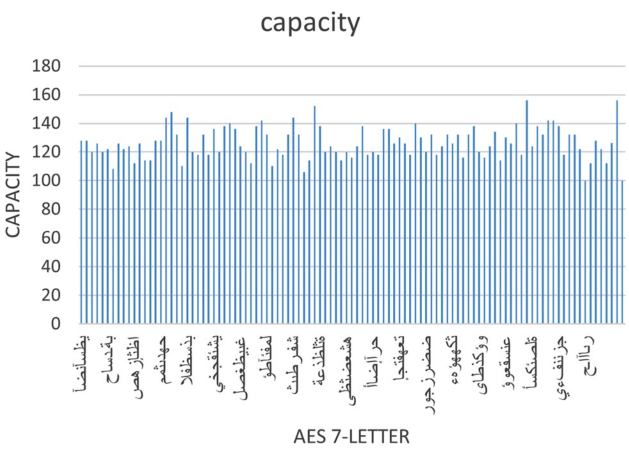


Fig. 17 Capacity for seven letters encrypted by AES

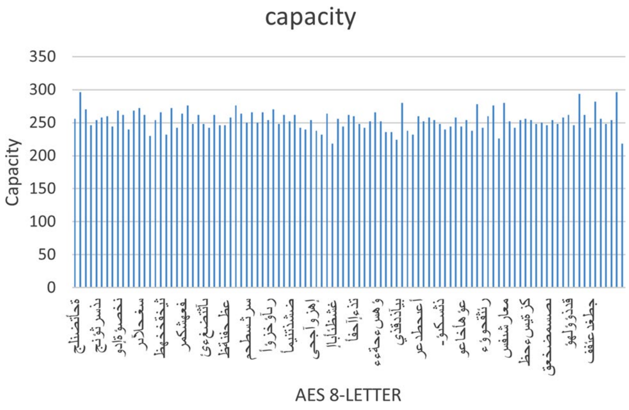


Fig. 18 Capacity for eight letters encrypted by AES

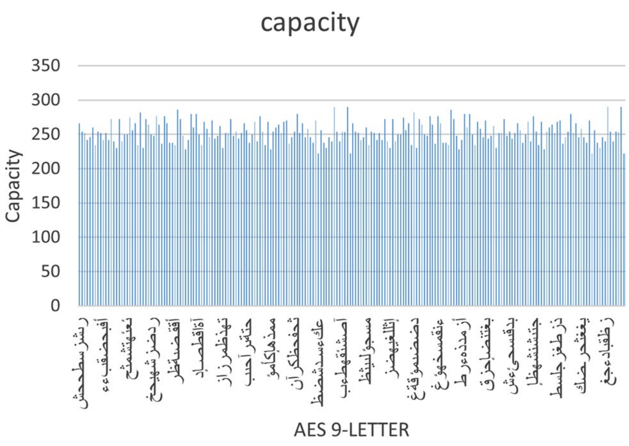


Fig. 19 Capacity for nine letters encrypted by AES

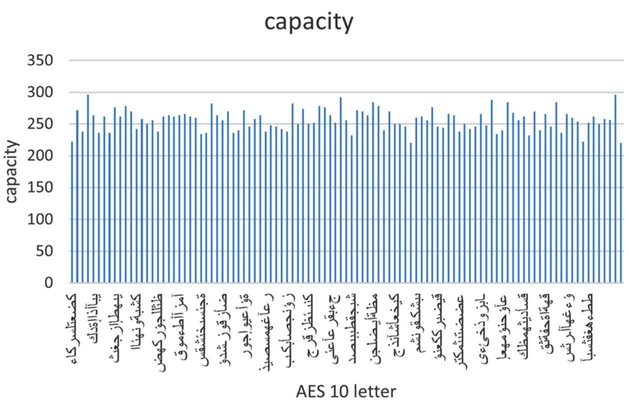


Fig. 20 Capacity for ten letters encrypted by AES

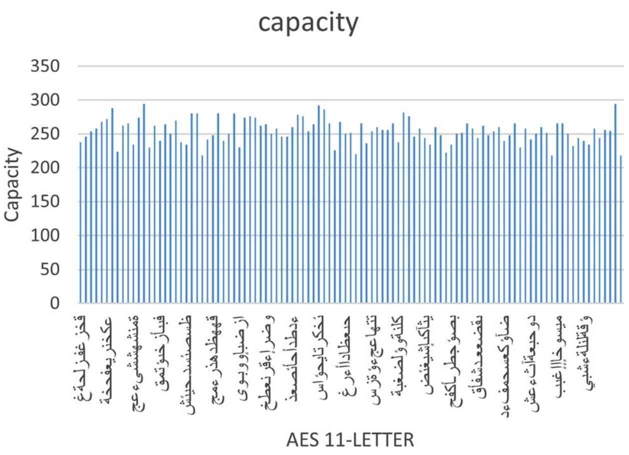


Fig. 21 Capacity for 11 letters encrypted by AES

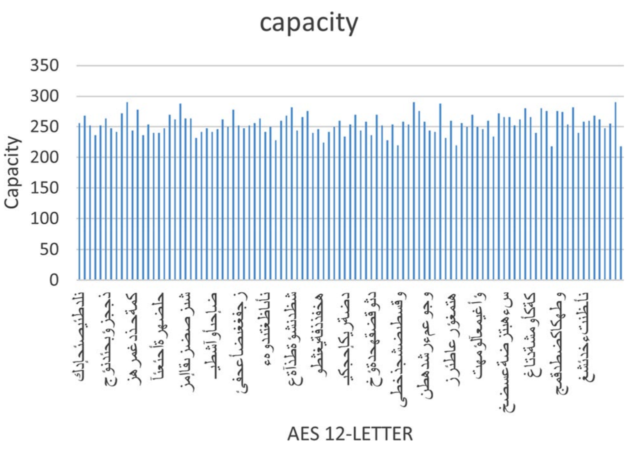


Fig. 22 Capacity for 12 letters encrypted by AES

the 100-names assuming different sizes and taking the averages from all. The figures are further showing some of the random names after intervals (as examples) to keep the reader in synchronization to the concept of randomness and it affects the linking the size of the secret data to the capacity measurements which is detailed as follows.

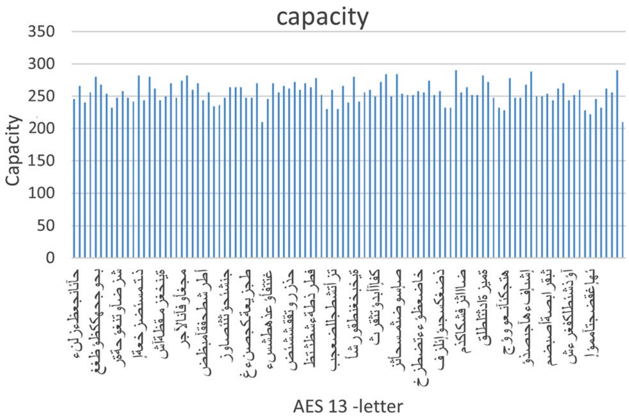


Fig. 23 Capacity for 13 letters encrypted by AES

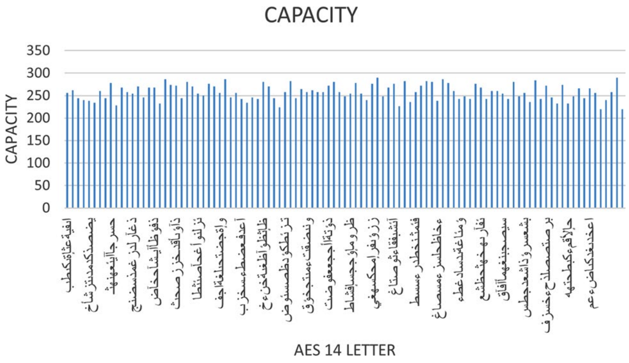


Fig. 24 Capacity for 14 letters encrypted by AES

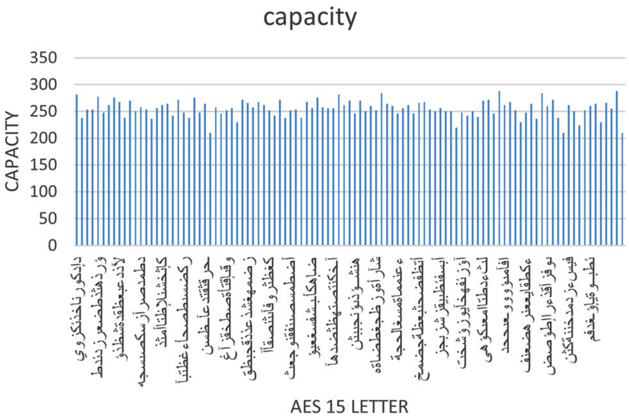


Fig. 25 Capacity for 15 letters encrypted by AES

For instance, consider Fig. 13, the largest value of the capacity is 158 linked to the name (غضء) where the lowest value is 102 when the name is (آها). The average capacity of the 100-names testing is 127. Similarly, in Fig. 14, the largest value of the 4-letter names capacity is 152 linked to the name (عن-ل) while the lowest value is 68 as for the name (لتصز). The average for 100 tests of the 4-letter

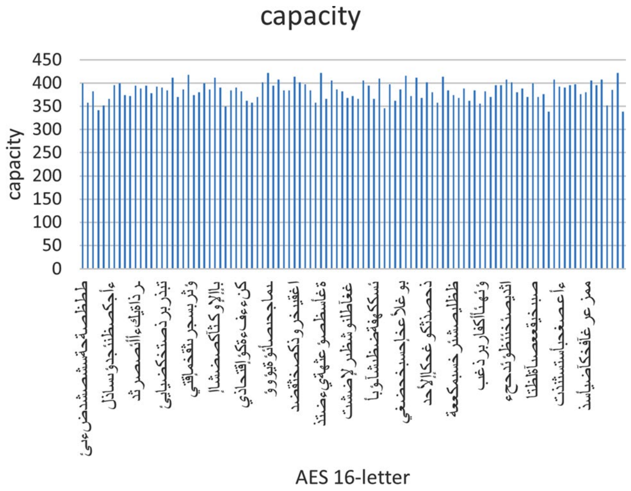


Fig. 26 Capacity for 16 letters encrypted by AES

names is found to be 126, which is slightly less than for 3-letters.

In Fig. 15, the largest value of the capacity is 156 connected to the name (ثإمخخ), while the lowest value is 98 linked to the name (خفزجة). The average for 100 tests of the 5-letter names is 127 similar to the 3-letters secret data. In Fig. 16, the largest value of the capacity is 156 for the name (ذوشظقط) while the lowest value is 108 with the name (خفزجة). The average for 100 tests of the 6-letters is 126 which is remarkably similar to the average of 4-letter names. In Fig. 17, the largest value of the capacity is 156 as for the name (طقملآيم), while the lowest value is 100 linked to the name (ح-اآاىر). The average for 100 tests of the 7-letters is 126. In Fig. 18, the largest value of the capacity is 296 as with the name (ضشرذثكع). The lowest value is 218 associated to the name (إابأئظشغ). The average for 100 tests of the 8-letter names is changing a lot to 254, starting a new range.

In Fig. 19, the largest value of the capacity is 290 with the name (أؤعيىثءيىي). The lowest value is 222 as for the name (ظضشد-سءكع). The average for 100 tests of the 9-letter names is 253, which can be considered logical related to the increase number of letters. In Fig. 20, the largest value of the capacity is 296 the name (ظزضظؤغةلدو), while the lowest value is 220 with the name (أكحدنن-س-فص). The average for 100 tests of the 10-letter names is 256, which is surprisingly less than for 9-letters but in the same range. In Fig. 21, the largest value of the capacity is 294 the name (ذغؤوأعتشزىغ), while the lowest value is 218 linked to the name (ءئأضجءةظجز). The average for 100 tests of the 11-letter names is 254, which is in the same range of 9-letters. In Fig. 22, the largest value of the capacity is 290 as for the name (ظوآطلنغءهاوى) while the lowest value is 218 linked to the name (اثلقشلكسسثثئ). The average for 100 tests of the 12-letter names is 255, interestingly similar to the range of 9-letters, 10-letters, and 11-letters. In Fig. 23, the largest value of the capacity is 290 the name (ةجوضعإتاذعجآب), while the lowest value is 210 the name (رغةصذرتممءؤطق). The average for 100 tests of the 13-letter names is 255. In Fig. 24, the largest value of the capacity is 290 with the name (ةجوضعإتاذعجآب), while the lowest value is 220 linked to the name (رغةصذرتممءؤطق). The average for 100 tests of the 14-letter names is 257. In Fig. 25, the largest value of the capacity is 288 linked to the name (شسهىئيملثغةضظدزأ) while the lowest value is 210 as of the name (جؤخ-ىئسغشدثتجلظد). The average for 100 tests of the 15-letter names is 255, which all in same range. In Fig. 26, the largest value of the capacity is 422 the name (ذتضءيةهثعؤصظسأغة), while the lowest value is 338 linked to the name (رنآه-ثيعلختحضقؤى). The average for 100 tests of the 16-letter names is 384, which is considered completely out of the range of recommendation.

Recall in this test, we selected 100 different names randomly fixing same size of characters to be used as sensitive data for the encryption crypto layer, i.e. by AES, DES, IDEA. The average capacity of AES experimentations is shown in Fig. 27 elaborating the minimum (MIN) and maximum (MAX) capacity sizes for verification purposes. The study capacity in general gave highest value using 16-char of value 422, and lowest value of 96 when the number is of 4-char. Interestingly, the AES capacity study can remark classification of three stages. The first stage of data secrets to be 3-char letters to 7-letters. The second stage is 8-letters to 15-letters. The final stage of 16-letters, which is completely not recommended to be used.

The same study of AES encryption is reapplied using DES and IDEA cryptography. For presentation limitations, the average results are reported. The changes in Fig. 28 is

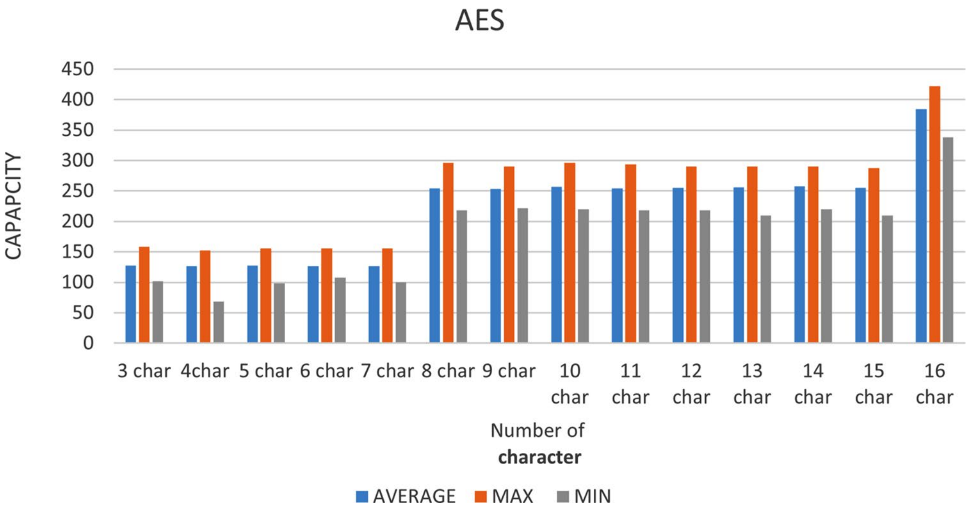


Fig. 27 Average AES capacity

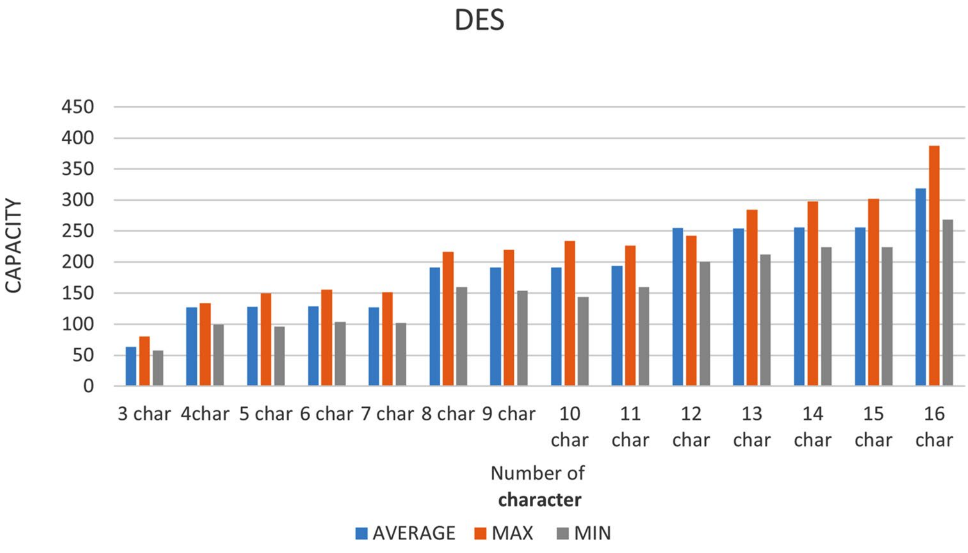


Fig. 28 Average DES capacity

the comparison of using MAX and MIN for every char. Observe that the stego layer with big difference of 16-char giving higher capacity change compared to the 3-char, which is expected. The capacity is low when 3-char encryption is used resulting in a capacity value of 58. When data are encrypted by IDEA, we get the highest value of the capacity as the number of characters is 15, the capacity is 280. The minimum value is when 4-characters are used showing the capacity value of 96, as shown in Fig. 29.

Capacity comparison of the two-layer system considering the three LWC: AES, IDEA, and DES, algorithms gave interesting feedback, as listed in Table 1. The work can give representative indication since our work used extensive testing of 100 secret randomly generated messages of lengths from 3-char letters to 16-char letters. The capacity comparison is visualized in Fig. 30 indicating the capacity preference to be increasing starting by DES then IDEA followed by AES. This study needs to be linked to the security

Fig. 29 Average IDEA capacity

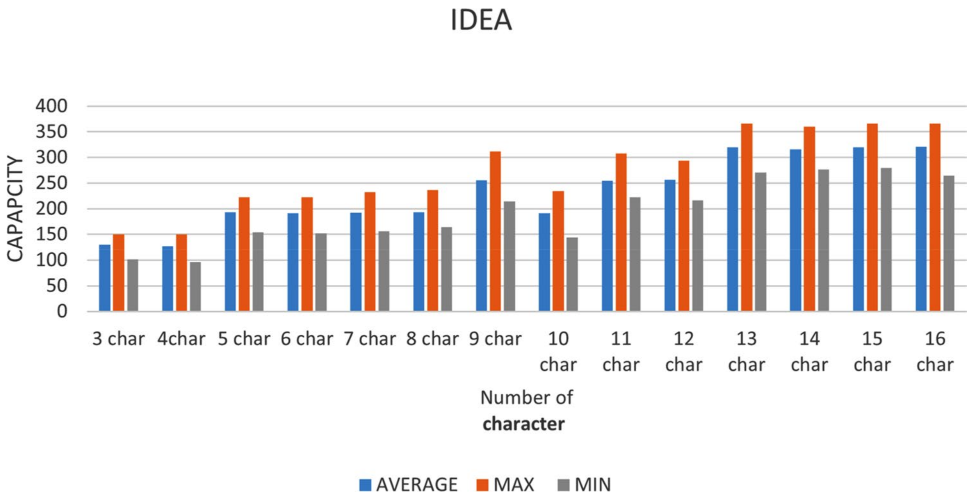
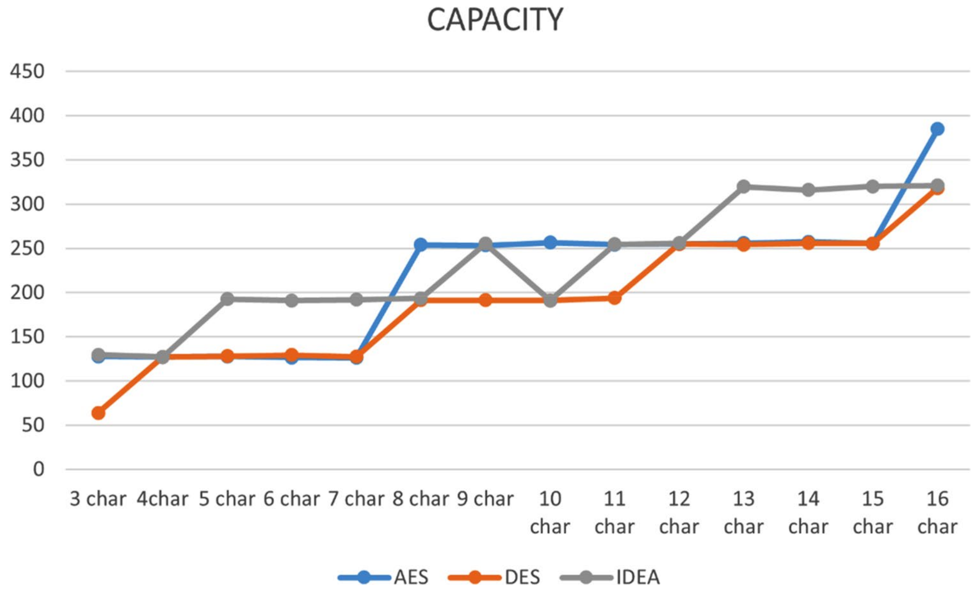


Table 1 System capacity comparison

Char	3	4	5	6	7	8	9	10	11	12	13	14	15	16
AES	127	127	127	126	126	254	253	256	254	255	255	257	255	384
DES	63	127	128	129	127	191	191	190	193	254	254	255	255	318
IDEA	129	127	192	90	191	193	255	190	254	255	319	316	320	321

Fig. 30 Capacity performance visual comparison



quantity measurements as well as security visual testing in order to highlight acceptable useful recommendations.

Security Quantity Testing

Security quantity testing refers to the ability of observers to notice that there are embedded data in the cover medium. This is measured by counting diacritics before and after concealment. The cover text, as shown in Fig. 11, contains 890 of diacritics, letters and spaces. The cover text contains 85 bytes of spaces which should be reduced providing:

$890 - 85 = 805.$

The diacritics and Arabic letters use 16-bits for each but the spaces represent one byte each for characters that must have a diacritics to hide. Thus, the number of letters equals the number of diacritics, making the cover-text number as: $805/2 = 402.5$

So, there are possibilities of almost 402 diacritics to hide. Since the diacritics represent 2 bytes, the amount of hiding possibilities is: $402 \times 2 = 804$ bytes.

When encoding with the AES algorithm, for example, cipher text length of 16 character, represented by 8 bits is: $16 \times 8 = 128$.

Therefore, we will get 128 binary bits, as of when experimenting with 100 names and taking the average of zero's to get 64 zeroes such as: $64 \times 2 = 128$ bytes.

The security testing study is considered to be comparing the diacritics before and after hiding. The study made a detailed study testing the 100 difference secret message ranging between 3-char letters and 16-char letters encrypted by AES, DES and IDEA, as results listed in Table 2.

Consider Table 2 to check the system security, observing the results of diacritics embedding, applying the two-layer system, the difference between the diacritics is interesting. Comparison before and after hiding is decreased and cannot be observed easily. The difference is so low such that no one can guess its usage in the information hiding process. Therefore, the security quantity test is supporting the capacity measurement in suggesting to use LWC: AES in the two-layer hiding information since the quantitative change in Arabic text is almost unnoticeable.

Security Visual Testing

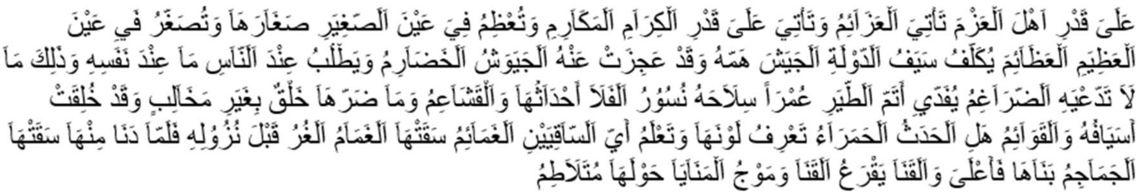
Security visual testing is used to measure homogeneity between the same two texts before and after concealment via peak signal-to-noise ratio (PSNR) measurements [29]. The PSNR represents a measure of the peak error, used to compute figure of merit of visual difference in decibels, i.e. between two images, used as quality visual measurement between the original and stego image. The higher the PSNR, the better the quality of the reconstructed image. This PSNR deals with images, therefore, we converted the text to image, read images of all different results and compare between the original cover-image and the stego-image, which contains the secret bits, similar to PSNR proof of research work in [30]. Figure 31 shows an examples of this intended image generated, i.e. as source of the PSNR original cover text, as of original Fig. 11, but removing all spaces.

To represent the idea, some AES study images are shown used as inputs examples for PSNR computations. For instance, Fig. 32 is showing cover-text image of one 3-letter secret data name encoded by AES algorithm, i.e. for the PSNR calculation.

Similarly, observe the image examples of text figures, i.e. Figs. 33, 34, 35, 36, 37, and 38, which shows image examples of hiding one 5-letter, 7-letter, 9-letter, 11-letter, 13-letter, and 15-letter names, respectively. Also, to apply this metric fairly, we used the hidden data and length but

Table 2 Security quantity comparison

Char	3	4	5	6	7	8	9	10	11	12	13	14	15	16
AES	676	676	676	678	678	550	550	548	550	548	548	546	548	420
DES	740	676	676	674	676	612	612	612	610	550	550	548	548	486
IDEA	674	676	612	614	612	610	548	548	550	548	484	488	484	482



عَلَىَ قَدْرِ اَهْلَ اَلْعَزْمَ تَأْتِيَ اَلْعَزَاَئِمُ وَتَأْتِيَ عَلَىَ قَدْرِ اَلْكِرَاَمِ اَلْمَكَاَرِمِ وَتُعْظِمُ فِيَ عَيْنَ اَلصَّغِيَرِ صَغَاَرَهَاَ وَتُصَغَّرُ فِيِ عَيْنَ
اَلْعَظِيَمِ اَلْعَظَاَئِمَ يُكَلِّفُ سَيْفُ اَلدَّوْلَةِ اَلْجَيْشَ هَمَّهُ وَقَدْ عَجِزَتْ عَنْهُ اَلْجَيُوشُ اَلْخِضَاَرِمُ وَيَطْلُبُ عِنْدَ اَلنَّاسِ مَاَ عِنْدَ نَفْسِهِ وَذَلِكَ مَاَ
لَاَ تَدَّعِيْهِ اَلضَّرَاَغِمُ يُفَدِّيِ أَتَمَّ اَلطَّيْرِ عُمْرَاً سِلَاَحَهُ نُسُوْرُ اَلْفَلَاَ أَحْدَاَثُهَاَ وَاَلْقَشَاَعِمُ وَمَاَ ضَرَّهَاَ خَلْقٌ بِغَيْرِ مَخَاَلِبٍ وَقَدْ خُلِقَتْ
أَسْيَاَفُهُ وَاَلْقَوَاَئِمُ هَلِ اَلْحَدَثُ اَلْحَمْرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أَيَّ اَلسَّاَقِيَيْنِ اَلْغَمَاَئِمُ سَقَتْهَاَ اَلْغَمَاَمُ اَلْغُرُّ قَبْلَ نُزُوْلِهِ فَلَمَّاَ دَنَاَ مِنْهَاَ سَقَتْهَاَ
اَلْجَمَاَجِمُ بَنَاَهَاَ فَأَعْلَىَ وَاَلْقَنَاَ يَقْرَعُ اَلْقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلَاَطِمِ

Fig. 31 The original (cover-text) as image for security analysis

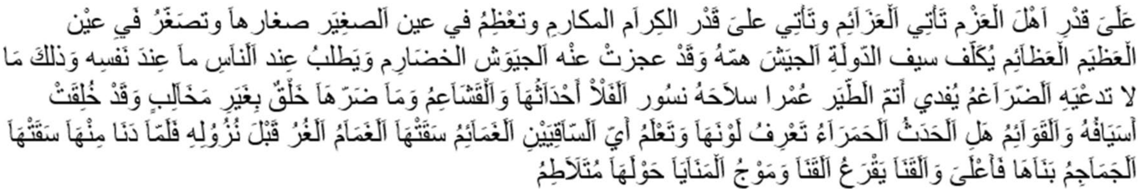

عَلَىَ قدرِ اَهْلَ اَلْعَزْم تَأْتِي اَلْعَزَاَئِم وتَأْتِي علىَ قَدْر الكِراَم المكارِمِ وتعْظِمُ في عين اَلصغِيَر صغارهاَ وتصَغَّرُ فيِ عيْن
اَلْعَظيَم اَلْعَظاَئِم يُكَلَف سيف الدّولَةِ اَلجيَشَ همّهُ وَقَدْ عجزتْ عنْه اَلجيَوَش الخضَارِم وَيَطلبُ عِند اَلْنَاسِ ماَ عِندَ نَفسِه وَذلكَ مَا
لا تدعْيَهِ اَلضَرَاَغمُ يُفدي أَتمَ الطّيَر عُمْرا سلاَحَهُ نسُور اَلْفَلاْ أَحْدَاَثُهَاَ وَاَلْقَشَاعِمِ وَمَاَ ضَرّهاَ خَلْقٌ بِغيَرِ مَخَالِبٍ وَقَدْ خُلِقَتْ
أَسَيَافُهُ وَاَلْقَوَاَئِمُ هَلِ اَلْحَدَثُ اَلْحمرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أَيَ اَلسّاَقِيَيْنِ اَلْغَمَاَئِمُ سَقَتْهَاَ اَلْغَمَاَمُ اَلْغُرُ قَبْلَ نُزُوْلِهِ فَلَمّاَ دَنَا مِنْهَاَ سَقَتْهَاَ
اَلْجَمَاَجِمُ بَنَاَهَا فَأَعْلَىَ وَاَلْقَنَاَ يَقْرَعُ اَلْقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلاَطِمُ

Fig. 32 Testing 3 char-text stego image

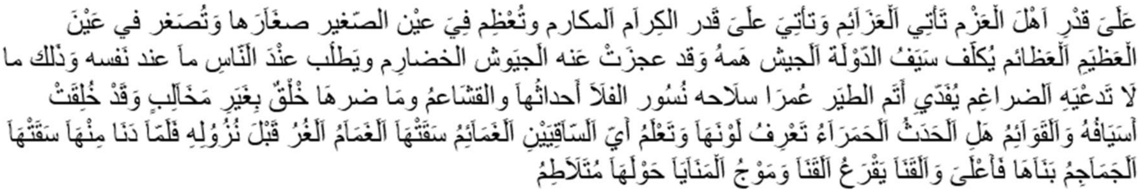

عَلَىَ قدرِ اَهْلَ اَلْعَزْم تَأْتِي اَلْعَزَاَئِم وَتَأْتِيَ علَىَ قَدر الكِراَم اَلمكارم وتُعْظِم فِيَ عيْن الصّغير صغَاَرَها وَتُصَغر في عَيْنَ
اَلْعَظيَمِ اَلْعَظاَئم يُكلَف سَيَفُ الدّوْلَة اَلجيش هَمهُ وَقد عجزَتْ عَنه اَلجيَوش الخضارِم ويَطلَب عنْدَ اَلنّاسِ ماَ عند نَفسه وَذْلك ما
لَا تدعْيَهِ اَلضراغِم يُفَدّيِ أَتم الطير عُمرَا سلاحه نُسُور الفلاَ أَحداثُهاَ والقشَاعمُ ومَا ضرها خُلْقٌ بِغيَرِ مَخَاَلِبٍ وَقَدْ خُلِقَتْ
أَسَيَافُهُ وَاَلْقَوَاَئِمُ هَلِ اَلْحَدَثُ اَلْحمرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أَيَ اَلسّاَقِيَيْنِ اَلْغَمَاَئِمُ سَقَتْهَاَ اَلْغَمَاَمُ اَلْغُرُ قَبْلَ نُزُوْلِهِ فَلَمّاَ دَنَا مِنْهَاَ سَقَتْهَاَ
اَلْجَمَاَجِمُ بَنَاَهَا فَأَعْلَىَ وَاَلْقَنَاَ يَقْرَعُ اَلْقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلاَطِمُ

Fig. 33 Testing 5 char-text stego image

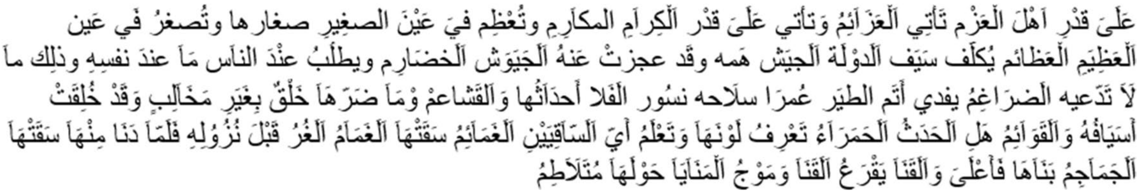

عَلَىَ قدرِ اَهْلَ اَلْعَزْم تَأْتِي اَلْعَزَاَئِمُ وَتَأْتي عَلَىَ قدْر اَلْكِراَمِ المكاَرِمِ وتُعْظِم فيَ عَيْنَ الصغيِرِ صغارها وتُصغرُ في عَين
اَلْعَظِيَمِ اَلْعَظائم يُكلَف سَيَف اَلدوْلَة اَلجيَش هَمه وقَد عجزتْ عَنهُ اَلْجَيَوَش اَلخضَارِم ويطلَبُ عنْدَ الناس مَاَ عندَ نفسِهِ وذلِك ماَ
لاَ تَدّعيه اَلضرَاَغِمُ يفدي أَتم الطيَر عُمرَا سلاحه نسُور اَلفَلا أَحداَثُها وَاَلقَشاعمْ وْمَاَ ضَرّهاَ خَلْقٌ بِغيَرِ مَخَاَلِبٍ وَقَدْ خُلِقَتْ
أَسَيَافُهُ وَاَلْقَوَاَئِمُ هَلِ اَلْحَدَثُ اَلْحمرَاَءُ تَعْرِفُ لَوْنَهَاَ وَتَعْلَمُ أَيَ اَلسّاَقِيَيْنِ اَلْغَمَاَئِمُ سَقَتْهَاَ اَلْغَمَاَمُ اَلْغُرُ قَبْلَ نُزُوْلِهِ فَلَمّاَ دَنَا مِنْهَاَ سَقَتْهَاَ
اَلْجَمَاَجِمُ بَنَاَهَا فَأَعْلَىَ وَاَلْقَنَاَ يَقْرَعُ اَلْقَنَاَ وَمَوْجُ اَلْمَنَاَيَاَ حَوْلَهَاَ مُتَلاَطِمُ

Fig. 34 Testing 7 char-text stego image

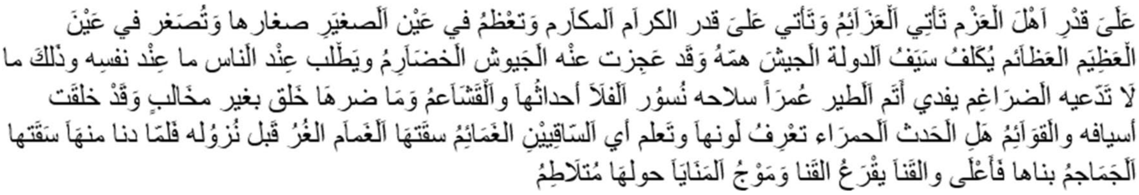

عَلَىَ قدرِ اَهْلَ اَلْعَزْم تَأْتِي اَلْعَزَاَئِمُ وَتَأْتي عَلَىَ قدر الكراَم اَلمكارم وَتعْظمُ في عَيْن اَلصغيَرِ صغارها وَتُصَغر في عَيْنَ
اَلْعَظِيَم العَظاَئم يُكلفُ سَيَفُ اَلدولة اَلجيشَ همّهُ وَقد عَجِزت عنْه اَلجَيوش اَلخضَارِمُ ويَطّلب عِنْد نفسِه وذْلكَ ما
لَا تَدّعيه اَلضرَاغِم يفدي أَتم اَلطير عُمرَاً سلاحه نُسُور اَلفَلاَ أَحداثُهاَ واَلْقَشَاعمُ وَمَا ضرها خَلق بغير مخالبٍ وَقَدْ خلقت
أسيافه واَلقَواَئِمُ هَلِ اَلحَدث اَلحمرَاء تعْرِفُ لَونهاَ وتَعلم أي اَلسّاَقِيِيْنِ الغَمَاَئِمُ سقتهَا اَلْغمام الغُرُ قَبل نُزُوله فلمّا دنا منهاَ سَقتها
اَلجَمَاجمُ بناها فأَعْلى والقناَ يقْرَعُ القنا وَموْجُ اَلمَنَاَياَ حولهَا مُتلاطِمُ

Fig. 35 Testing 9 char-text stego image

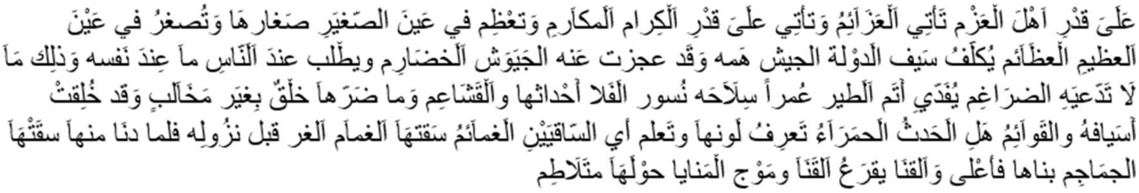
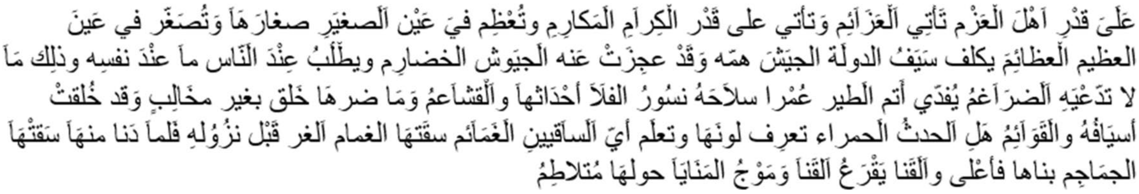

عَلَىَ قدرِ اَهْلَ اَلْعَزْم تَأْتِي اَلْعَزَاَئِمُ وَتَأْتِي علَىَ قدْرِ اَلْكِرام اَلمكارِمِ وَتعْظِم في عَينَ الصّغيَرِ صَغارهَا وَتُصغرُ في عَيْنَ
اَلعظيِمِ اَلْعظَاَئم يُكَلّفُ سَيف اَلدّوْلة الجيش هَمه وَقد عجزت عَنه الجَيَوَش اَلخضَارِم ويطّلب عندَ اَلنّاسِ ماَ عِندَ نَفسه وَذلِك مَاَ
لَا تَدّعيَهِ الضرَاغِم يُفَدّيِ أَتم اَلطير عُمراً سِلاَحَه نُسور اَلفَلا أَحْداثها واَلْقَشَاعِم وَما ضَرّهاَ خلْقٌ بِغيَر مَخَاَلبٍ وَقد خُلِقتْ
أَسَيافهُ والقَوائِمُ هَلِ اَلحَدثُ الحمرَاَءُ تَعرِفُ لَونها وتَعلم أي السّاَقِيَيْنِ اَلغمائمُ سَقتهَاَ اَلغمامَ اَلغر قبل نزُوله فلما دنَا منهاَ سقتْهَاَ
الجمَاجِم بناها فأعْلى وَاَلقنَا يقرَعُ اَلقنَاَ ومَوْج اَلْمَنايا حوْلهَاَ متَلاطِم

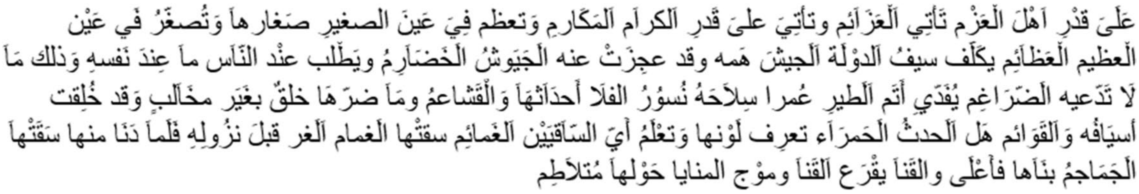
Fig. 36 Testing 11 char-text stego image

via encryption of DES and IDEA then hidden in stego text providing the average PSNR security visual testing results, as listed in Table 3. The results are showing interestingly similar (within same range) values indicating that all three methods can be considered acceptable from security visual comparisons point of view, which is a completely different



عَلَىَ قدْرِ اَهْلَ اَلعَزْم تَأتِي اَلعَزَاَئِم وَتأتي على قَدْر اَلكِراَمِ اَلمَكارِمِ وتُعْظِم فيَ عَيْن اَلصغيَرِ صغارَهَاَ وَتُصَغّر في عَينَ
العظيم اَلعظائِمَ يكلف سَيَفُ الدولَة الجيَشَ همّه وَقَدْ عجزَتْ عَنه اَلجيَوش الخضارِم ويطّلبُ عِنْدَ النّاس ماَ عنْدَ نفسِه وذلِك مَاَ
لا تدّعْيَهِ اَلضرَاَغمُ يُفدّي أتم اَلطير عُمْرا سلاَحَهُ نسُورُ الفلاَ أحْدَاثها واَلْقشاَعمُ وَمَا ضرهَا خَلق بغير مخَالِبٍ وَقد خُلقتْ
أسيَاَفُهُ والقَوَاَئِمُ هَلِ اَلحدثُ اَلحمراء تعرِف لونَهَا وتعلَم أيَ اَلساقيينِ اَلغَمَاَئم سقَتهَا الغمام اَلغر قَبْل نزُوُلهِ فلماَ دَنا منهَاَ سَقتْهَاَ
الجمَاجِم بناها فأعْلى واَلقَنا يَقْرَعُ اَلقناَ وَمَوْجُ المَنَايَاَ حولهَا مُتلاطِمُ

Fig. 37 Testing 13 char-text stego image



عَلَىَ قدْرِ اَهْلَ اَلعَزْم تَأتِي اَلعَزَاَئِم وتأتِيَ علىَ قدرِ اَلكراَم اَلمَكَارِمِ وَتعظم فِيَ عَينَ الصغيرِ صَغارها وَتُصغّرُ فَي عَيْنِ
اَلعظيم اَلعَظاَئِم يكَلّف سيفُ اَلدوْلَة اَلجيش هَمه وقد عجزَتْ عنه اَلجَيوشُ اَلخَضاَرِمُ ويَطّلب عنْد النّاَس ماَ عِند نَفسهِ وَذلك مَاَ
لَاَ تَدّعيه اَلضّرَاَغِم يُفَدّي أَتم اَلطيرِ عُمرا سِلاَحَهُ نُسوُرُ الفلاَ أحدَاَثَهاَ وَاَلْقشاعمُ ومَاَ ضرّهَا خلقٌ بغَيَر مخَاَلبٍ وَقد خُلِقت
أسيَاَفُه وَاَلقَواَئم هَل اَلحدثُ اَلحَمرَاَء تعرِف لَوْنها وَتعْلَمُ أيَ السّاَقيَيْن اَلغمائم سقتْها اَلغمام اَلغر قبلَ نزُولهِ فَلَماَ دَنَا منها سَقتْها
اَلجَمَاجمُ بنَاَها فأعْلَى والقَنا يقْرَع اَلقناَ وموْج المنايا حَوْلهاَ مُتلاَطِم

Fig. 38 Testing 15 char-text stego image

Table 3 Security visual comparison (PSNR testing)

Char	3	4	5	6	7	8	9	10	11	12	13	14	15	16
AES	14.5	11.8	14.3	12.2	13.5	12.1	12.7	12.1	11.8	12.5	12.7	11.4	12.5	12
DES	13.3	12.2	11.8	11.8	13.1	11.8	11.8	11.9	15	12.3	12.9	11.8	13.2	12
IDEA	12.1	11.7	11.4	12.4	12.6	12.3	13	12.1	11.7	11.6	11.5	12.6	11.5	12

research work and remarks than the presentation of Al-Otaibi [13] and the first Arabic Kashida utilization [31] security system.

Conclusion

In this work, we have shown how to design the two-layer Arabic text security system for hiding sensitive text data on limited capability devices. We used two-layers of cryptography and steganography, utilizing steganography in increasing security level similar in principle to the counting-based secret-sharing scheme [32], which was originally presented in [33]. The crypto work made this multimedia stego data hiding completely different than [34] in its interesting combination between steganography layer and cryptography layer.

The cryptography layer is adopted to ensure independent security while steganography layer is fully dependent on the user and his data available. The system has been implemented on visual studio platform showing interesting results as intelligent tuning of previous research of image steganography two-layer [13] and three-layer [14] which involved "heavy-weight" crypto algorithms dedicated completely for different research objective. This research worked on securing data via lightweight cryptography and Arabic text steganography trying to select benefits from both cryptography and security intended to run over limited capability devices, i.e. providing acceptable security.

The research run performance analysis defined over the specific scope related to background works presented in [10–14]. As this research is serving limited capability real-life user mobile device needs, as part of proof of concept, the research figured out that the percentage of embedding as well as the security is not changing much by changing the single layer tools alone, as serving applicable solution to be used. Therefore, the work tested hiding texts with different sizes that gave interesting remarks. The study simulations have been tested many times on different randomly selected Arabic texts finding coherent analogous results presented proofing the concept and building trust in the explanations. This has been set, within the scope of specific work at this stage, as samples to show LWC steganography enhancement effects in relation to this idea of work as well as revisiting comparable approaches studied. The system steganography layer embedded data in the Arabic text using lightweight cryptography (LWC) algorithms, AES, DES, and IDEA. The study tested its implementation assuming different secret data sizes ranging from 3-letters to 16-letters. Every letter option is run on 100 random number tests showing interesting feedback.

The research indicated that DES had higher capacity than AES and IDEA algorithms, providing acceptable security, serving our intention within this work to build up to be used as references to compare with all coming improvements and related researches. The work experimentation showed the results to be classified into three stages based on their similarities. It proved that as the number of Arabic characters increases or decrease within the same range, the preference number of hiding parameters is common. Therefore, the diacritics (to hide encrypted data) decreases in the text showing high capacity. On the other hand, AES proofed higher security than DES and IDEA under optical standard but is not considered highly efficient and not recommended to be used in limited device situations.

Acknowledgements The authors like to thank Umm Al-Qura University for all their motivation and support towards this research.

Funding There is no funding source.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Agreement We declare that this work is original and not was considered to be published in any other publication media.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Kheshaifaty N, Gutub A. Preventing multiple accessing attacks via efficient integration of captcha crypto hash functions. Int J Comput Sci Netw Secur. 2020;20(9):16–28. https://doi.org/10.22937/IJCSNS.2020.20.09.3.
2. Almutairi S, Gutub A, Al-Ghamdi M. Image steganography to facilitate online students account system. Rev Bus Technol Res. 2019;16(2):43–9.
3. Bin-Hureib E, Gutub A. Enhancing medical data security via combining elliptic curve cryptography and image steganography. Int J Comput Sci Netw Secur. 2020;20(8):1–8. https://doi.org/10.22937/IJCSNS.2020.20.08.1.
4. Gutub A. utilizing information security techniques as digital evidence for cybercrime activities. In: Cybercrimes and Digital Forensics Forum, Naif Arab University for Security Sciences (NAUSS), 2019. http://doi.org/10.13140/RG.2.2.14885.45281
5. Almutairi S, Gutub A, Al-Juaid N. Motivating teachers to use information technology in educational process within Saudi Arabia. Int J Technol Enhanced Learn. 2020;12(2):200–17. https://doi.org/10.1504/IJTEL.2020.10027118.
6. Dinu D, Corre Y, Khovratovich D, Perrin L, Großschädl J, Biryukov A. Triathlon of lightweight block ciphers for the internet of things. J Cryptogr Eng. 2019;9:283–302.
7. Alassaf N, Gutub A. Simulating light-weight-cryptography implementation for IoT healthcare data security applications. Int J E-Health Med Commun. 2019;10(4):1–15. https://doi.org/10.4018/IJEHMC.2019100101.
8. Alanazi N, Khan E, Gutub A. Functionality-improved Arabic text steganography based on unicode features. Arab J Sci Eng. 2020. https://doi.org/10.1007/s13369-020-04917-5.
9. Bailey K, Curran K. An evaluation of image based steganography methods. Multimed Tools Appl. 2006;30:55–88. https://doi.org/10.1007/s11042-006-0008-4.
10. Gutub A, Ghouti L, Elarian Y, Awaideh S, Alvi A. Utilizing diacritic marks for Arabic text steganography. Kuwait J Sci Eng. 2010;37(1):89–109.
11. Gutub A, Al-Nazer A. High capacity steganography tool for Arabic text using “Kashida.” ISC Int J Inf Secur. 2010;2(2):107–18. https://doi.org/10.22042/ISECURE.2015.2.2.4.
12. Odeh A, Alzubi A, Hani Q, Elleithy K. Steganography by multipoint Arabic letters. In: IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2012, pp. 1–7.
13. Al-Otaibi N, Gutub A. 2-leyer security system for hiding sensitive text data on personal computers. Lect Notes Inf Theory. 2014;2(2):151–7. https://doi.org/10.12720/lnit.2.2.151-157.
14. Samkari H, Gutub A. Protecting medical records against cybercrimes within Hajj period by 3-layer security. Recent Trends Inf Technol Appl. 2019;2(3):1–21.
15. Alassaf N, Gutub A, Parah S, AlGhamdi M. Enhancing speed of SIMON: a light-weight-cryptographic algorithm for IoT applications. Multimed Tools Appl. 2019;78:32633–57. https://doi.org/10.1007/s11042-018-6801-z.
16. Alanazi N, Khan E, Gutub A. Efficient security and capacity techniques for Arabic text steganography via engaging unicode standard encoding. Multimed Tools Appl. 2020. https://doi.org/10.1007/s11042-020-09667-y.
17. Al-Otaibi N, Gutub A. Flexible stego-system for hiding text in images of personal computers based on user security priority. In: International conference on advanced engineering technologies (AET), 2014; pp. 250–256.
18. Al-Nofaie S, Fattani M, Gutub A. Capacity improved Arabic text steganography technique utilizing ‘Kashida’ with whitespaces. In: International conference on mathematical sciences and computer engineering (ICMSCE2016), 2016; pp. 38–44.
19. Kadhem SM, Ali DW. Proposed hybrid method to hide information in Arabic text. J Theor Appl Inf Technol. 2017;95(7):1466–1478.
20. Malik A, Sikka G, Verma HK. A high capacity text steganography scheme based on LZW compression and color coding. Eng Sci Technol. 2017;20:72–9.
21. Mohamed A. An improved algorithm for information hiding based on features of Arabic text: a unicode approach. Egypt Inform J. 2014;15:79–87.
22. Bensaad ML, Yagoubi MB. High capacity diacritics-based method for information hiding in Arabic text. In: International conference on innovations in information technology. 2011; pp. 433–436.
23. Al-Haidari F, Gutub A, Al-Kahsah K, Hamodi J. Improving security and capacity for arabic text steganography using ‘Kashida’ extensions. In: IEEE/ACS international conference on computer systems and applications. 2009; pp. 396–399.
24. Al-Azawi AF, Fadhil MA. Arabic text steganography using Kashida extensions with Huffman code. J Appl Sci. 2010;10:436–9.
25. Shirali-Shahreza M, Shirali-Shahreza S. High capacity Persian/Arabic text steganography. Appl Sci. 2008;8:4173–9.
26. Por LY, Wong K, Chee KO. UniSpaCh: a text-based data hiding method using unicode space characters. J Syst Softw. 2012;85:1075–82.
27. Desoky A. Listega: list-based steganography methodology. Int J Inf Secur. 2009;8:247–61.

28. Alabish A, Goweder A, Enakoa A. A universal lexical steganography technique. Int J Comput Commun Eng. 2013;2:153–157. https://doi.org/10.7763/IJCCE.2013.V2.159.
29. Al-Nofaie S, Gutub A. Utilizing pseudo-spaces to improve Arabic text steganography for multimedia data communications. Multimed Tools Appl. 2020;79:19–67. https://doi.org/10.1007/s11042-019-08025-x.
30. Al-Nofaie S, Gutub A, Al-Ghamdi M. Enhancing Arabic text steganography for personal usage utilizing pseudo-spaces. J King Saud Univ Comput Inf Sci. 2019. https://doi.org/10.1016/j.jksuci.2019.06.010.
31. Gutub A, Fattani M. A novel Arabic text steganography method using letter points and extensions. Int J Comput Electr Autom Control Inf Eng. 2007;1(3):502–5. https://doi.org/10.5281/zenodo.1061621.
32. AlKhodaidi T, Gutub A. Refining image steganography distribution for higher security multimedia counting-based secret-sharing. Multimed Tools Appl. 2020. https://doi.org/10.1007/s11042-020-09720-w.
33. Gutub A, Al-Juaid N, Khan E. Counting-based secret sharing technique for multimedia applications. Multimed Tools Appl. 2019;78:5591–619. https://doi.org/10.1007/s11042-017-5293-6.
34. Hassan F, Gutub A. Efficient reversible data hiding multimedia technique based on smart image interpolation. Multimed Tools Appl. 2020;79(39):30087–109. https://doi.org/10.1007/s11042-020-09513-1.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.