Journal of Engineering and Computer Sciences
Qassim University, Vol. 10, No. 2, pp. 105-124 (July 2017/Shawwal 1438H)

A Pedagogical Multi-Key Multi-Stage Package to Secure Communication Channels

Ali Muhammad Ali Rushdi and Ahmed Abdullah Alsheikhy

Department of Electrical and Computer Engineering,

Faculty of Engineering, King Abdulaziz University,

P. O. Box 80204, Jeddah, 21589, Saudi Arabia

arushdi@kau.edu.sa

Abstract. Information security has many aspects such as authentication, privacy, integrity and non-repudiation. This paper studies implementation of information security through cryptography and steganography. Cryptography can be loosely defined as the scrambling of data so that only an intended authorized user can unscramble it. Cryptography allows secure transfer of information over space (transmission over a communication channel) or over time (storage within a computer memory). Steganography is the related concept of hiding some piece of information within another piece of information (possibly of another type), such that the hidden information would be accessibly only to some intended authorized users. This paper offers a tutorial overview of cryptography and steganography from the perspective of a multi-key multi-stage software package (named *EDSP* for *Encryption Decryption Software Package*). This package serves as a pedagogical aid accompanying this current tutorial, though it is, admittedly, neither highly advanced nor particularly novel. It employs a one-time pad technique, wherein a key is used only once. The *EDSP* combines a multitude of strong cryptographic algorithms, secret (symmetric) key ones together with public (asymmetric) key ones including the well-known RSA and El-Gamal algorithms, whose cryptanalyses show their relative unbreakability in front of strong attacks. Depending on the number of algorithms used, low, medium and high levels of security are achieved. Several random keys are generated and used only once through any encryption or decryption process. Depending on the size of message being encrypted, these keys are used through several schemes which are determined by the user. The package supports Arabic and English texts and can be enhanced to include more languages. It can be used with a variety of platforms with no specific hardware required. The package has a built-in capability of converting images into text and vice -verse, a key step in utilizing its cryptographic features for steganographic purposes. The steganography method is basically applied in the EDSP through converting any image into a text file or vice versa. However, hiding a secret image into another image is also performed and provided in the package. The paper briefly overviews the mathematics of pertinent operations, describes the capabilities of the developed package, and demonstrates its usefulness via illustrative examples. As an offshoot, the paper highlights contributions of the Arabs (during the glorious Islamic era) to cryptography, in general, and to cryptanalysis, in particular.

Keywords**:** El-Gamal cryptosystem, RSA algorithm, Cryptography, Steganography, Information Security, Communication and Storage, Multi-Key Multi-Stage Package, EDSP, Arab contributions.

1. Introduction

Cryptography is the science of using mathematics to encrypt and decrypt data. It allows secure transfer of information over space (transmission over a communication channel) or over time (storage within a computer memory). While cryptography is the science of securing data, *cryptanalysis* is the science of analyzing and breaking secured communication. Classical *cryptanalysis* involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck [1]. Until modern times cryptography referred almost exclusively to *encryption*, which is the process of converting ordinary information (*plaintext*) into unintelligible gibberish (i.e., *ciphertext*). *Decryption* is the reverse, in other words, moving from the unintelligible *ciphertext* back to *plaintext*. A *cipher* (or cypher) is a pair of *algorithms* which create the *encryption* and the reversing *decryption*. The detailed operation of a *cipher* is controlled both by the *algorithm* and in each instance by a *key*. This is a secret parameter (ideally known only to the communicants) for a specific message exchange context. Keys are important, as ciphers without variable keys can be trivially broken with only the knowledge of the cipher used and are therefore less than useful for most purposes.

Cryptography can be directly used to help ensure certain security properties, including:

✓ ***Authentication***: the process of proving one's identity, which is sometimes split into two types, namely, (1) *Entity authentication* (Each of the communicating parties verifying that the other parties are the right ones), and (2) *Data-origin authentication* (Who is claiming to be the sender of the message really is the one from whom it originates).

✓ ***Integrity***: the process of preventing unauthorized modification of messages or data. This guarantees that the content of the message, that was sent, has not been tampered with.

✓ ***Confidentiality (Privacy)***: the process of ensuring that no one can read the message except the intended receiver.

✓ ***Non-repudiation***: the process of ensuring that the message has been sent and received. Non-repudiation of origin protects against denial by one of the entities involved in a communication of having participated in all or part of the communication. *Non-repudiation with proof of origin* protects against any attempts by the sender to refuse to acknowledge having sent a message, while *non-repudiation with proof of delivery* protects against any attempt by the recipient to deny, falsely, having received a message.

Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication. There are, in general, three types of cryptographic schemes typically used to accomplish these goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions [2-5].

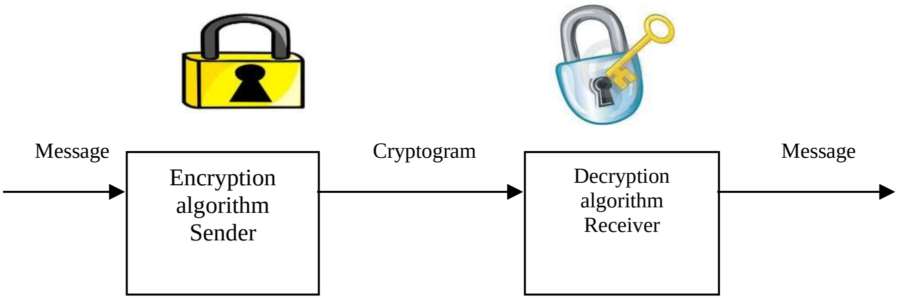


Fig. (1). Elements of a cryptography system.

Figure 1 illustrates the main elements of a cryptography system. In this figure, three terms are noted, namely, **encryption** (the process of scrambling data according to a specific algorithm), **decryption** (the process of decoding data to its original form, so that it can be easily read by the intended receiver) and **key** (a sequence of symbols represented in binary form as a string of bits to control the mechanism of encryption and decryption algorithm) [1-5].

Until a few decades ago, cryptography was largely of interest to the military and the diplomats. Nowadays, it permeates many aspects of life, like Internet banking, e-commerce, e-mail, and automatic teller machines. It has tremendous impact on the economic, sociological, and political aspects of the contemporary society. Recently, cryptography has been utilized as a pedagogical tool in getting students interested in mathematics, by motivating them, giving them good reasons to spend time on a subject that requires hard work, and, if possible, involving them in research [6].

Related to the science and art of cryptography is the one of steganography. Cryptography and steganography are considered cousins in the spycraft family [7-12]. Cryptography scrambles a message so it cannot be understood. Steganography hides or conceals the message so that its mere existence cannot be detected or even suspected. A message in ciphertext, for instance, might arouse suspicion on the part of the recipient, while an “invisible” message created with steganographic methods will not [7].

We digress here a little bit to include a brief historical note on cryptography. The earliest forms of secret writing required little more than local pen and paper analogs, as most people could not read. More literacy required more advanced cryptography. The main classical cipher types are (a) *transposition ciphers*, which rearrange the order of letters in a message and (b) *substitution ciphers*, which systematically replace letters or groups of letters with other letters or groups of letters. The earliest known use of cryptography is some carved ciphertext on stone in Egypt (1900 BC). The next oldest is bakery recipes from Mesopotamia (present-day

Iraq). The next development was a substitution cipher due to the Roman Emperor, Julius Caesar. The credit for the earliest cryptanalysis technique, namely, frequency analysis, is certainly due to the Muslim Arabs. Kahn the famous historian of cryptography, equivocally asserts [13] that "Cryptography was born among the Arabs" and that " (they) were the first to discover and write down the methods of cryptanalysis". Many people attribute the discovery of cryptanalysis to the Arab philosopher Al-Kindi (known in the west as Alkindus) but actually cryptanalysis is a contribution of an earlier genius polymath, namely, Al-Khalil Ibn Ahmad Al-Farahidi [14-16]. Further information on the history of cryptography can be found in the lucid text of Kahn [13], and in the lucid survey papers [17-22].

This paper offers a tutorial overview of cryptography and steganography from the perspective of a multi-key multi-stage software package. The package employs a one-time pad technique, wherein a key is used only once. The package combines a multitude of strong cryptographic algorithms, secret (symmetric) key ones together with public (asymmetric) key ones such as the RSA [23] and El-Gamal algorithms [24-27], whose cryptanalysis shows their relative unbreakability in front of strong attacks, provided they are supported by some preprocessing [28-30]. The RSA and El-Gamal schemes are used in the EDSP to provide very secure and powerful tools with other approaches such as Addition, Subtraction, Multiplication and XORing for achieving high level of desired security. Depending on the number of algorithms used, low, medium and high levels of security are achieved. The package supports Arabic and English texts and can be enhanced to include more languages. The package can be used with a variety of platforms with no specific hardware required. The package has a built-in capability of converting images into text and vice versa, a key step in utilizing its cryptographic features for steganographic purposes.

The rest of this paper is organized as follows: Section 2 briefly discusses the basic operations used in cryptography and further offers a brief introduction to the RSA and El-Gamal algorithms. Section 3 presents the system requirements, types of users and a glimpse of operations for the software package developed herein. For future reference, we name this package the "***Encryption and Decryption Software Package (EDSP)***." Section 4 presents some typical results obtained by this package as a way of demonstrating certain features of cryptography and steganography. Section 5 concludes the paper.

2. Basic Operations

In the EDSP system, many operations are involved in order to ensure high strength of cryptography and steganography. We use Addition, Subtraction, Multiplication, Swapping, Inserting, Circular shift, Exclusive OR and in particular, we use the RSA and El-Gamal algorithms. All operations used herein are implemented in a technique called *One-Time Pad* [31] so that a key used is randomly generated only once even if the operation is repeated many times. The package provides multi keys and multi stages based on the number of operations. In addition, two special operations namely Steganography and converting image into text and vice versa are used.

2.1 The RSA Algorithm

RSA is the best known public-key algorithm, named after its inventors: *Rivest, Shamir and Adleman* [23]. **RSA** uses public and private keys that are functions of a pair of large prime numbers. Its security is based on the difficulty of factoring a composite integer that equals the product of two large primes. The **RSA** algorithm can be used for both public key encryption and digital signatures. The keys used for encryption and decryption in the **RSA** algorithm, are generated using random data. The key used for encryption is a public key and the key used for decryption is a private key [28-30]. Public keys are stored anywhere publicly accessible. The sender of a message encrypts the data using the public key, and the receiver decrypts it using the private key generated by the sender [28-30]. That way, no one else can intercept the data except the receiver. The **RSA** algorithm involves three steps: ***key generation, encryption and decryption.***

Key generation: **RSA** involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key [28-30]. The keys for the **RSA** algorithm are generated as follows [23]:

- Choose two distinct prime numbers p and q. In general, p and q are very large numbers, for security purposes, the integers p and q should be chosen uniformly at random and should be of similar bit-length.

- Compute $n = p*q$, where n is used as the modulus for both the public and private keys.

- Compute $\varphi(pq) = (p - 1)(q - 1)$.

- Choose an integer **e** such that $1 < e < \varphi(pq)$, and e and $\varphi(pq)$ share no divisors other than 1 (i.e. e and $\varphi(pq)$ are *co-prime or relatively prime*). Now e is released as the public key exponent and choosing **e** having a short addition chain results in more efficient encryption. Small public exponents (such as $e = 3$) could potentially lead to greater security risks.

- Determine d (using modular arithmetic) which satisfies the congruence relation: $de = 1 \ (mod\ \varphi(pq))$.

The public key consists of the modulus **n** and the public (or encryption) exponent **e**. The private key consists of the modulus n and the private (or decryption) exponent d which must be kept secret.

Encryption: this step can be done with the following procedures:

- Obtain authentic public key (n,e).

- Represent the message as an integer m in the interval [0,n-1].

- Compute c = m^e mod n where the value of c represents the ciphertext.
- Send the ciphertext to the recipient.

Decryption: to decrypt message m from ciphertext, use private key d as follows:

- m = c^d mod n.

Example

Let p= 2357, q = 2551 so n = p*q = 2357*2551 = 6012707 and Φ = (p-1)*(q-1) = 6007800. Select e = 3674911, using extended Euclidean algorithm to get that d = 422191, the public key is the pair (n = 6012707, e = 3674911) where the private key is d = 422191. To encrypt a message m = 5234673 → c = m^e mod n = 5234673^3674911 mod 6012707 = 3650502. Send c to the recipient and to decrypt m from ciphertext → m = c^d mod n. So m = 3650502^422191 mod 6012707 = 5234673.

RSA Encryption in Practice

✓ Recommended size of modulus (n) > 1024 (binary bits).

✓ Selecting primes: Roughly same size p and q to prevent elliptic curve factoring.

✓ p and q should be large enough.

2.2 El-Gamal Algorithm

El-Gamal encryption system is an asymmetric key encryption algorithm for public key cryptography which is based on the **Diffie-Hellman** paradigm [31]. It was described by ***Taher El-Gamal*** in 1985 [24, 25]. Essentially, it generates a shared secret key and uses it as a one-time pad to encrypt one block of data. ***El-Gamal*** is the predecessor of **DSS** and is perfectly usable today, although no widely known standard has been created for it. **El-Gamal encryption** consists of three components: *the key generator, the encryption algorithm, and the decryption algorithm.*

KEY GENERATOR: this step can be done as follows:

1. Select large prime number p.
2. Select random number g in the interval [1, p -1].
3. Select private key number a.
4. Compute value of y = g^a mod p, so the public key is (p,g,y) where the private key is a.

ENCRYPTION: to achieve this procedure, the sender must do the following:

1. Select random integer number k in the interval [0, p -1].
2. Represent the message m as an integer in the range [0, p -1].

3. Compute first part of elgamal block y1 = g^k mod p.

4. Compute second part of elgamal block y2 = m*y^k mod p.

5. Combine both results to appear as one block which is the ciphertext and send it to the recipient.

DECRYPTION: to decrypt a desired message from ciphertext, do the following:

1. Obtain sender private key a.
2. Compute m = (y2* (y1^a)^-1) mod p, the value of m is the original message.

EXAMPLE

Let p = 23 and choose g = 11, select a = 6 so y = g^a mod p = 11^6 mod 23 = 9. Now the public key is (23,11,p) and private key is 6. Select k = 3 and m = 10, compute y1 = g^k mod p = 11^3 mod 23 = 20, y2 = (m*y^k) mod p = (10*9^3) mod 23 = 22, thus the message m converted into ciphertext of (20,22). To decrypt ciphertext again compute m = (y2*(y1^a)^-1) mod p = (22* (20^6)^-1) mod 23 = 10.

***Efficiency* of El-Gamal encryption**

✓ The encryption process requires two modular exponentiations, namely g^k mod p and y^k mod p. These exponentiations can be sped up by selecting random exponent k having some additional structure, for example, having low Hamming weights.

✓ A disadvantage of El-Gamal encryption is that there is message expansion by a factor of 2. That is, the ciphertext is twice as long as the corresponding plaintext.

Security of El-Gamal encryption

The problem of breaking the El-Gamal encryption scheme, i.e., recovering m given p, g, y, y1 and y2 is equivalent to solving the Diffie-Hellman problem. In fact, the El-Gamal encryption scheme can be viewed as simply comprising a Diffie-Hellman key exchange to determine a session key g^a*k and then encrypting the message by multiplication with that session key. For this reason, the security of the El-Gamal encryption scheme is said to be based on the discrete logarithm problem.

It is critical that different random integers k be used to encrypt different messages. Suppose the same k is used to encrypt two messages m1 and m2 and the resulting ciphertext pairs are (y11; y12) and (y21; y21). Then y12/y21 = m1/m2, and m2 could be easily computed if m1 were known.

Attacks on the RSA or El-Gamal cryptosystems

It has not been shown that a devastating attack can be conducted in a reasonable amount of time on a correctly implemented RSA or El-Gamal cryptosystem . However, there are many successful attacks on poor or naïve implementations. Boneh et al. [29] presented an attack on plain El-Gamal and plain RSA encryption. The attack showed that without proper preprocessing of the plaintexts, both El-Gamal and RSA encryptions are fundamentally insecure. Namely, when one uses these systems to encrypt a (short) secret key of a symmetric cipher it is often possible to recover the secret key from the ciphertext. These results demonstrated that preprocessing messages prior to encryption should be an essential part of successful implementations of both systems. Therefore, our EDPS system strengthens its implementation of the combined RSA and El-Gamal cryptosystems by utilizing multi-keys, multi-stages, and a variety of supplementary operations. In passing, one should stress that the advantages acquired by preprocessing the plaintext before encryption are achieved at the cost of increasing the side information that have to be sent to the receiver for decryption.

Other operations are performed using a letter by letter method. Iinitially, several random integer values are randomly generated, and then, several random keys are determined and generated based on the size of the message being processed. Each letter is associated with a unique key. A defined operation is then performed to generate the ciphertext. Based on the user's choice, 5 or 7 or 9 operations take place in different levels of security in the text mode while different numbers of operations are performed in the image mode. The RSA approach is an essential operation in each level whereas the El-Gamal scheme is performed only in the highest level to produce a powerful and unpredictable ciphertext.

3. The Encryption-Decryption Software Package (EDSP)

In this section, we briefly describe the characteristics and operation of the Encryption-Decryption Software Package (EDSP). We discuss the EDSP system requirements, types of users and operation guide.

3.1 System Requirement

☒ Pentium, Pentium Pro, Pentium II, Pentium III, Pentium IV, or AMD Athlon based personal computer.

☒ Microsoft Windows 95, Windows 98 (original and Second Edition), Windows Millennium Edition (ME), Windows NT 4.0 (with Service Pack 5 for Y2K compliancy or Service Pack 6a), or Windows 2000.

☒ CD-ROM drive (for installation from CD).

☒ 64 MB RAM minimum, 128 MB RAM recommended.

☒ Disk space varies depending on size of partition and installation of online help files. The MathWorks Installer will inform you of the hard disk space requirement for your particular partition.

☒ 8-bit graphics adapter and display (for 256 simultaneous colours). Other recommended items include:

☒ Microsoft Windows supported graphics accelerator card, printer, and sound card.

☒ Microsoft Word 7.0 (Office 95), 8.0 (Office 97), or Office 2000 is required to run the MATLAB Notebook.

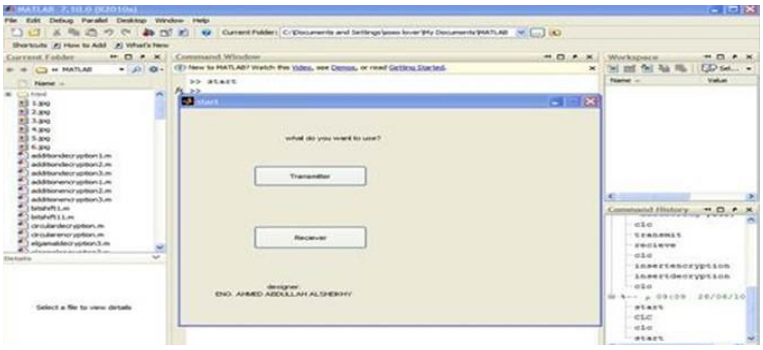
☒ MATLAB can be set up to operate on a network via the TCP/IP communications protocol.

3.2 Types of Users

There are two types of users who can use or run EDSP, namely: *An administrator user* can apply the tasks of (a) Adding a new user, (b) Deletion of an existing user, and (c) Starting an Encryption or Decryption process. *A Regular user* can apply only one task which is to start an Encryption or Decryption job.

3.3 A glimpse of EDSP operation

We now give a glimpse of operations of the tasks common to administrator and regular users, namely that on MATLAB command window, type (***start***) to run the program. The first screen of EDSP (Fig. 2) asks the user which mode of operation is intended to be used and performed. The user selects either the transmitter or receiver mode by clicking on it and a new log-in window will appear asking for username and password (Fig. 3). After entering a valid username and correct password a new window will appear prompting the user to determine the type of input and output data format, (text or image) (Fig. 4). After that, the user is required (Fig. 5) to select a level of security among three choices (low, medium and high), where every level has its own number of operations (5 (2) operations for low level, 7 (4) operations for medium level and 9 (5) operations for high level in the text (image) format), respectively.


MATLAB 7.10.0 (R2010a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Documents and Settings\[illegible]\My Documents\MATLAB
Current Folder
Command Window
Workspace
Command History
start
Transmitter
Receiver
Select a file to view details

```
>> start
>>
```

what do you want to use?

designer:
ENG. AHMED ABDULLAH ALSHEIKHY

```
clc
transmit
recieve
clc
insertencryption
insertdecryption
clc
start
clc
clc
start
```

Fig. (2). The first screen of EDSP.

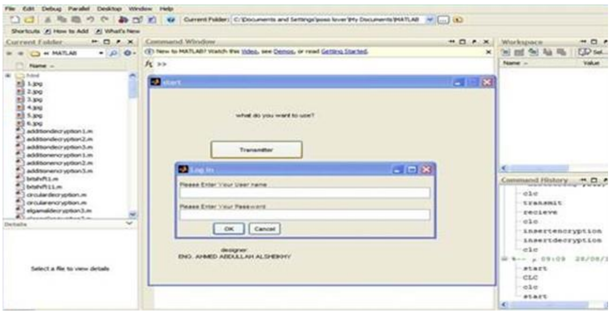


Fig. (3). The log-in screen.

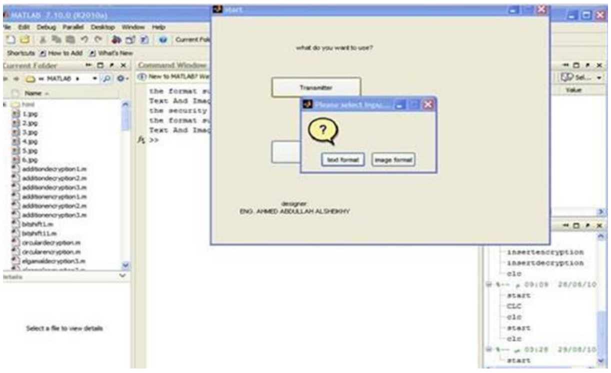


Fig. (4). Selection of input or output format.

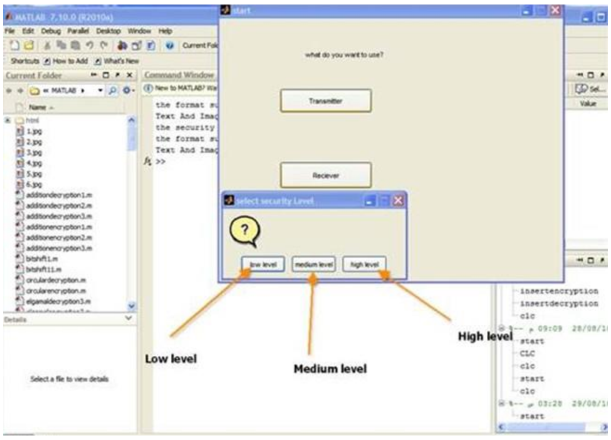


Fig. (5). Selection of security level.

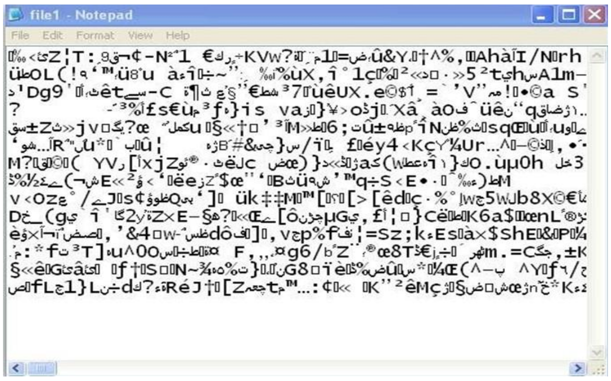
4. Results

Example 1

Suppose that we want to encrypt the following Arabic text, taken from [13]

بدأت معالجة التعمية باعتبارها علماً مع نهاية القرن الثامن ميلادي أو القرن الأول هجري. ابتدأت هذه الحقبة بالخليل بن أحمد الفراهيدي, وابن كيسان, وابن وحشية النبطي, وأبي حاتم السجستاني, وتُوجِت بعمل يعقوب الكندي في القرن الثالث الهجري \ التاسع الميلادي, الذي أوفى فيه على الغايةِ دقة وشمولاً وتحليلاً وتصنيفاً واستعمالاً لخواص اللغةِ التي يُعمَّى أو يُحل بها, واستمرت هذه الحقبة حتى تاريخنا المعاصر متراوحةً بين خُمودٍ وازدهارٍ. فقد بدأت تخفتُ بعد عصرِ الكندي إلى أن أتت هجمات المغول وحملات الصليبيين, فازدهرت من جديد في القرن السابع والثامن الهجريين \ الثالث عشر والرابع عشر الميلاديينِ, فكثرت الكتب المصنّفة فيها على أيدي ابن دُنينير وابنِ عدلانَ وابنِ الدُريهم وغيرهم.

The encrypted text file is given by Fig. 6.



```
▯‰<ئZ¦T:ﯛق¬¢-N²1 €ركـ÷KVw?▯ـم1▯=ضû&Y.▯†^%,▯AhàÏI/N▯rh
üطOL(!٩'™ü8u à؛î▯÷~'' %▯%ùX,î°1ç▯%▯²«د»▯·»5²tيshسA1m–
د'Dg9'▯ثêtـے-C ة¶ث ع§''€شط³7▯ùêUX.e©$1 = `'V''مه!▯•©a S'
? ‐`³%£s€ù³f}is vaز▯}¥›oڈj▯ˆXâ àOف^üê''qﮔضاق١ژ..
سق±Zث»jv▯يگ?œ ˝كملu▯§«†▯'³ÏM»ط▯6;تû±ظهpîNظن%ث▯sqŒ▯ù▯uوا▯ے
'شو…ÎR"ںu*▯`ب▯û¦ ژB´#&چی}س/ï▯ £▯éy4<KçY¼Ur…^▯–©▯,•
M?▯ق▯©▯( YV,[ÏxjZئو®·ثëJc ضœ)}د«▯ژa ک(Wعطة î١}كO.ùµ0h خل3
ڈ%½ع ے(¬شE«²ؤ<'▯ëezZ$œ''▯Bث ü٩ش'™q÷S<E•·▯˜‰)طM
v<Ozع°/ےJ▯s¢ؤظوQبی']▯ ük‡‡M▯™[▯▯[>[êd▯ç·%°Jwج5WJb8X©€ﻛ
Dخ_(gي î'گ2yةZ×E–§ه?▯«Œے[ôچژنµGي,£ا¦▯}Cë▯ط▯K6a$▯œnL®ژ
èؤ×Ï¬ اصض▯,'&4▯w-ظس dôف▯]▯,vجp%fئ¦=Sz;k؛Es▯àx$ShE▯&▯P▯¼
م:*fت³T]uٔ^00س▯÷ط▯ة¤ F,,,¤g6/bZ ˝®œ8Tڈ€j÷▯ شهرm.=Cجگ,±K
§«ê▯Gئâ▯ئ ▯f†▯S▯▯N~¾ه٥%ت}▯▯ںG8▯ïè▯%ضûس▯*▯¼Œ(^–پ ^Y▯f٦/ج
ص▯fLج1}Lن÷dك?ةRéJ†▯[Zچعمt™…:¢▯« ▯K''²êMçژ▯§ض▯شœژnخ*K؛ڈ
```

Fig. (6). Encrypted text for Example 1.

Example 2

Suppose that we want to encrypt the following English text, taken from [13]:

Cryptology was born among the Arabs. They were the first to discover and write down the methods of cryptanalysis. The people that exploded out of Arabia in the 600s and flamed over vast areas of the known world swiftly engendered one of the highest civilizations that history -had yet seen. Science flowered. Arab medicine and mathematics became the best in the world—from the latter, in fact, comes the word "cipher." Practical arts flourished. Administrative techniques developed. The exuberant creative energies of such a culture, excluded by its religion from painting or sculpture, and inspired by it to an explication of the Holy Koran, poured into literary pursuits. Storytelling, exemplified by Sheherazade's *Thousand and One*

Nights, word-riddles, rebuses, puns, anagrams, and similar games abounded; grammar became a major study. And included was secret writing.

The encrypted text file is given by Fig. 7.

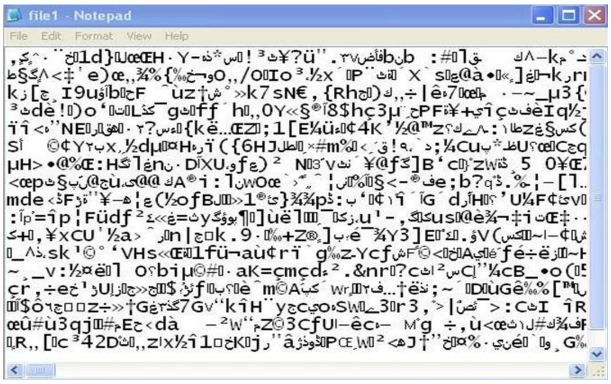
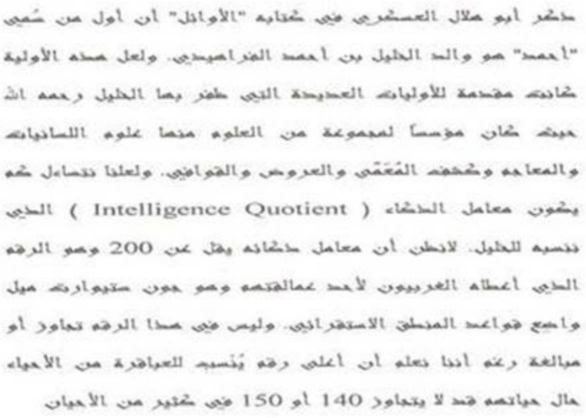


Fig. (7). Encrypted text for Example 2.

Example 3

Suppose that we want to use the airplane image given by Fig. 8 as a cover image to hide the text image of Fig. 9 inside it. Our EDPS system produces the stego image of Fig. 10, in which the text of Fig. 9 is hidden in an image looking exactly alike the one in Fig. 8. Later, our system manages to recover the hidden text in Fig. 11. Note that in the EDSP, the stego image is the same size as hidden image. In this paper, the stego image is shrinked due to space limitation.

Fig. (8). Cover image for Example 3.



ذكر أبو هلال العسكري في كتابه "الأوائل" أن أول من سُمي
"أحمد" هو والد الخليل بن أحمد الفراهيدي. ولعل هذه الأولية
كانت مقدمة للأوليات العديدة التي ظفر بها الخليل رحمه الله
حيث كان مؤسساً لمجموعة من العلوم منها علوم اللسانيات
والمعاجم وكشف المُعَمّى والعروض والقوافي. ولعلنا نتساءل كم
يكون معامل الذكاء (Intelligence Quotient) الذي
ننسبه للخليل. لانظن أن معامل ذكائه يقل عن 200 وهو الرقم
الذي أعطاه الغربيون لأحد عباقرتهم وهو جون ستيوارت ميل
واضع قواعد المنطق الاستقرائي. وليس في هذا الرقم تجاوز أو
مبالغة رغم أننا نعلم أن أعلى رقم يُنسب للعباقرة من الأحياء
حال حياتهم قد لا يتجاوز 140 أو 150 في كثير من الأحيان

Fig. (9). Text image to be hidden in Example 3.

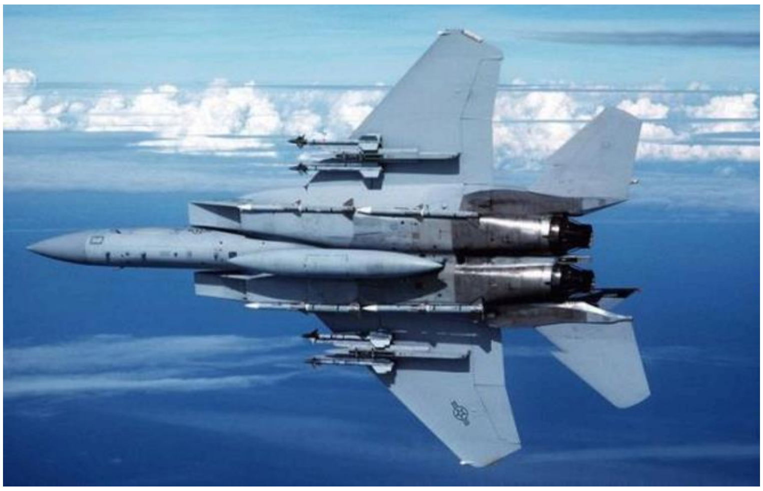
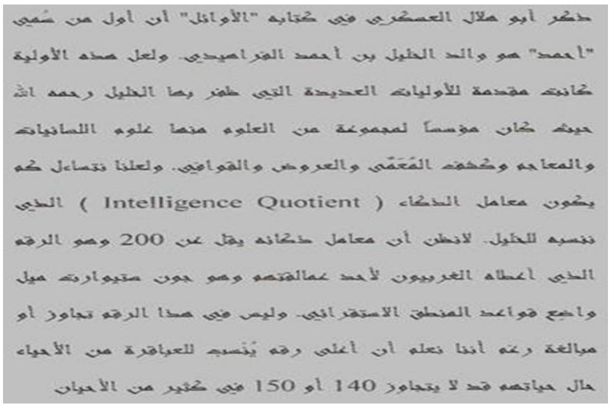


Fig. (10). Stego image for Example 3.

The stego image looks exactly like the cover image, so no one can think or predict that there is a hidden text composed within it.



ذكر أبو هلال العسكري في كتابه "الأوائل" أن أول من سُمي
"أحمد" هو والد الخليل بن أحمد الفراهيدي. ولعل هذه الأولية
كانت مقدمة للأوليات العديدة التي ظفر بها الخليل رحمه الله
حيث كان مؤسساً لمجموعة من العلوم منها علوم اللسانيات
والمعاجم وكشف المُعَمّى والعروض والقوافي. ولعلنا نتساءل كم
يكون معامل الذكاء (Intelligence Quotient) الذي
ننسبه للخليل. لانظن أن معامل ذكائه يقل عن 200 وهو الرقم
الذي أعطاه الغربيون لأحد عمالقتهم وهو جون ستيوارت ميل
واضع قواعد المنطق الاستقرائي. وليس في هذا الرقم تجاوز أو
مبالغة رغم أننا نعلم أن أعلى رقم يُنسب للعباقرة من الأحياء
حال حياتهم قد لا يتجاوز 140 أو 150 في كثير من الأحيان

Fig. (11). Recovered image for Example 3.

Example 4

Suppose that we want to use the flowers image given by Fig. 12 as a cover image to hide the map image of Fig. 13 inside it. Our EDPS system produces the stego image of Fig. 14, in which the map of Fig. 13 is hidden in an image looking exactly alike the one in Fig. 12. Later, the developed system manages to recover our hidden map in Fig. 15.

Fig. (12). Cover image for Example 4.

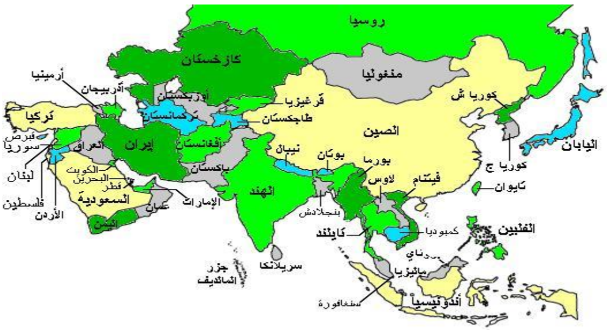


Fig. (13). Map image to be hidden in Example 4.

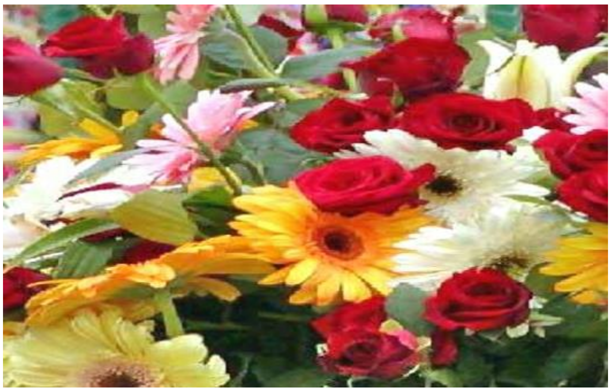


Fig. (14). Stego image for Example 4.

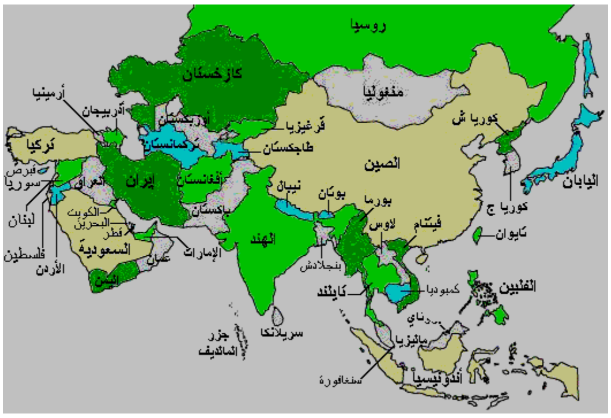


Fig. (15). Recovered image for Example 4.

5. Conclusions

Encryption allows the protection of sensitive data files from unauthorized users. This paper offers a tutorial of mathematical, historical, and elementary implementation aspects in the perspective of a simple package called the Encryption and Decryption Software Package (EDSP). Despite the simplicity of the EDSP package, its implementation, features, and typical results capture many of the essential features of the cryptography paradigm. Hence, this package serves as a pedagogical aid accompanying this current tutorial, rather than as an advanced novel tool competing with available commercial packages. As a bonus, the EDSP system can be used as a demonstrative tool for steganography as well as cryptography. The EDSP system scrambles data files to make sure that they cannot be read by individuals who do not have the key to decrypt the files. The most effective encryption algorithms being used herein are the El-Gamal cryptosystem algorithm and the RSA algorithm. The EDSP system is intended to satisfy user's requirements, emphasize one-time-pad only and ensure the high strength of encryption. The algorithm used herein uses a multiple of unpredictable operations in the encryption process, and it also employs a key that is a sequence of unknown operations. The unique features of this system are: the ability to generate very different keys for every single message even if it is sent more than once, and the insertion algorithm which will scatter the data in a matrix containing random numbers which will increase the size and length of the original data and make it more difficult to distinguish the original message from it. The EDSP system is implemented in the MATLAB language. It can be designed using any other language but MATLAB was chosen since it has many supporting tools like dealing with matrices, mathematical operations, logical operations, movies, images and sounds. The EDSP system combines many prominent algorithms in composite algorithms of various degrees of strength. Correctness of the algorithms is demonstrated via some working examples.

The EDSP system is believed to be reasonably secure against exhaustive brute force attacks or cryptanalysis. This belief stems from the fact it had been made extremely hard for an adversary to discover the pertinent algorithms and stages, the real keys, or the sequence of the operations used. However, the EDSP system is definitely not unbreakable. The point of its breakability can be decided via formal cryptanalysis or attack by a competent adversary.

6. Acknowledgment

The authors are indebted to Dr. Ahmad Ali Rushdi of ALFARABI Consulting Group, California, USA for the help he kindly offered in the preparation of this manuscript.

7. References

[1] **Tanenbaum**, **A. S**., and **Wetherall**, **D. J**., *Computer Networks*, Fifth Edition, Pearson Education, Boston, MA, USA, (2011).

[2] **Menezes**, **A.**, **Oorschot**, **P.** and **Vanstone**, **S.**, *Handbook of Applied Cryptography*, CRC Press Company, New York, NY, USA, (1997).

[3] **Adler**, **M.** and **Gailly**, **J. L.**, *An Introduction to Cryptography*, Network Associates, Santa Clara, CA, USA, (1999).

[4] **Piper**, **F**., *Cryptography*, John Wiley & Sons, New York, NY, USA, (2002).

[5] **Curry**, **I.,** *An Introduction to Cryptography and Digital Signatures*, *Entrust Securing Digital Identities and Information*, USA, (2001).

[6] **Kaur, M**., "Cryptography as a Pedagogical Tool," *PRIMUS: Problems, Resources, and Issues in Mathematics Undergraduate Studies*, Vol. 18, No. 2, (2008), pp. 198-206.

[7] **Johnson**, **N. F**., and **Jajodia**, **S**., "Exploring steganography: Seeing the unseen." *IEEE Computer*, Vol. 31, No. 2, (1998), pp. 26-34.

[8] **Anderson**, **R. J**., and **Petitcolas**, **F. A. P**. , “On the limits of steganography,” *IEEE Journal of selected Areas in Communications*, Vol. 16, No. 4, (1998), pp. 474-481.

[9] **Artz**, **D.**, "Digital steganography: hiding data within data." *IEEE Internet Computing*, Vol. 5, No. 3 (2001), pp. 75-80.

[10] **Al-Barhmtoshy**, **H**., **Osman**, **E.**, and **Ezzat**, **M**., "A novel security model combining cryptography and steganography." *Proceedings of the Seventeenth National Computer Conference*, Al-Madinah Al-Munw'warah, Saudi Arabia (2004), pp. 483-491.

[11] **Khan**, **F.**, and **Gutub**, **A. A**., "Message Concealment Techniques using image-based Steganography," *The 4th IEEE GCC Conference and Exhibition*, (2007).

[12] **Manoj**, I. V. S., "Cryptography and Steganography," *International Journal of Computer Applications*, Vol. 1, No. 12, (2010), pp. 63-68.

[13] **Kahn**, **D**., *The Codebreakers*: *The Story of Secret Writing*, The Macmillan Company, New York, NY, USA (1967).

[14] **Mrayati**, **M.**, **Meer Alam**, **Y.** and **Al-Tayyan**, **H.**, *Origins of Arab Cryptography and Cryptanalysis*, Vol. I: *Analysis and Editing of Three Arabic Manuscripts*, Arab Academy of Damascus Press, Damascus, Syrian Arab Republic, In Arabic (1987).

[15] **Mrayati**, **M.**, **Meer Alam**, **Y.** and **Al-Tayyan**, **H**., *Origins of Arab Cryptography and Cryptanalysis*, Vol. II : *Analysis and Editing of Eight*

Arabic Manuscripts, Arab Academy of Damascus Press, Damascus, Syrian Arab Republic, In Arabic (1996).

[16] **Mrayati, M., Meer Alam, Y.,** and **Al-Tayyan, H.,** *Arabic Origins of Cryptology*, KFCRIS & KACST, Riyadh, Saudi Arabia, 6 volumes, (2004).

[17] **Al Kadi**, **I. A**., "Origins of cryptography: The Arab contributions," *Cryptologia*, Vol. 16, No. 2, (1992), pp. 97-126.

[18] **Massey, J. L.,** "Review of Series on Arabic Origins of Cryptology," *Cryptologia*, Vol. 32, No. 3, (2008), pp. 280-283.

[19] **Schwartz, K. A.,** "Charting Arabic Cryptology's Evolution," *Cryptologia*, Vol. 33, No. 4, (2009), pp. 297–304.

[20] **Azizi, A.** and **Azizi, M.,** "Instances of Arabic Cryptography in Morocco," *Cryptologia*, Vol. 35, No. 1, (2011), pp. 47-57.

[21] **Azizi, A.** and **Azizi, M.,** "Instances of Arabic Cryptography in Morocco II," *Cryptologia*, Vol. 37, No. 4, (2013), pp. 328-337.

[22] **Schwartz, K. A.,** "From Text to Technological Context: Medieval Arabic Cryptology's Relation to Paper, Numbers, and the Post," *Cryptologia*, Vol. 38, No. 2, (2014), pp. 133-146.

[23] **Rivest**, **R. L.** , **Shamir**, **A**., and **Adleman**, **L.**, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, Vol. 21, (1978), pp. 120-130.

[24] **El-Gamal**, **T**., "A public key cryptosystem and a signature scheme based on discrete logarithms." *Advances in Cryptology*, Lecture Notes in Computer Science, Vol. 196, Springer, Berlin-Heidelberg, Germany, (1985), pp. 10-18.

[25] **El-Gamal**, **T**., "A public key cryptosystem and a signature scheme based on discrete logarithms", *IEEE Transactions on Information Theory*, Vol. IT-31, No. 4, (1985), pp. 469-479.

[26] **Hwang**, **M. S., Chang**, **C. C**., and **Hwang**, **K. F**., "An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering,* Vol. 14, No. 2, (2002), pp. 445-446.

[27] **Meier**, **A. V**., *The El-Gamal Cryptosystem*, CRC Press, New York, NY, USA, (2005).

[28] **Boneh**, **D**., "Twenty years of attacks on the RSA cryptosystem," *Notices of the American Mathematical Society (AMS),* Vol. 46, No. 2 (1999), pp. 203-213.

[29] **Boneh**, **D**., **Joux**, **A**., and **Nguyen**, **P. Q.,** "Why textbook El-Gamal and RSA encryption are insecure." *Advances in Cryptology—ASIACRYPT 2000*, Springer, Berlin-Heidelberg, Germany, (2000), pp. 30-43.

[30] **Pellegrini**, **A.**, **Bertacco**, **V.** and **Austin**, **T.**, *"Fault-based attack of RSA authentication*," Proceedings of the Conference on Design, Automation and Test in Europe, European Design and Automation Association, Leuven, Belgium, Belgium (2010), pp. 855-860.

[31] **Bellovin**, **S. M.,** "Frank Miller: Inventor of the One-Time Pad," Cryptologia, Vol. 35, No. 3, (2011), pp. 203-222.

[32] **Diffe**, **W.**, and **Hellman**, **M.**, "New directions in cryptography," *IEEE Transactions on Information Theory*, Vol. 22, No. 6, (1976), pp. 644-654.

حزمة برمجية تعليمية عديدة المفاتيح وعديدة المراحل لتأمين قنوات الاتصال

علي محمد علي رشدي أحمد عبدالله الشيخي

قسم الهندسة الكهربائية وهندسة الحاسبات، كلية الهندسة،

ص. ب. ٨٠٢٠٤، جامعة الملك عبد العزيز, جدة، ٢١٥٨٩، المملكة العربية السعودية

arushdi@kau.edu.sa

مخلص البحث. ثمة جوانب عديدة لأمن المعلومات منها التوثيق والخصوصية والصواب وعدم الجحود. تدرسُ الورقة هذه تحقيق أمن المعلومات من خلال التعمية والتخفية. يُمكِن إعطاء تعريف فضفاض للتعمية بأنها خلط البيانات بحيث تتعذر استعادتها على غير المستخدِم المقصود المصرَّح له. تسمح التعمية بالنقل الآمن للمعلومات عبر المكان (كما في الانتقال عبر قنوات الاتصالات)، وعبر الزمان (كما في التخزين بذاكرات الحاسبات). أما التخفِية فتمثل مفهوماً وثيق الصِلة يتعلق بإخفاء بعض المعلومات داخل البعض الآخر (الذي ربما يكون مُختلِفاً عنه في النوع)، بحيث يتعذر الوصول إلى المعلومات المخفاة لغير المستخدِمين المقصودين المصرَّح لهم. تقوم الورقة بمراجعة تعليمية لمفاهيم التعمية والتخفية من منظور حِزمة برمجية مُتعدّدة المفاتيح ومُتعدّدة المراحل (تسمى اختصارابالأحرف *ح ب ع ك* التي ترمز إلى حزمة برمجية للتعمية وكشف المعمى). تمثل هذه الحزمة أداة توضيحية تدعم الشرح التعليمي المقدم هنا رغم كونها ليست شديدة التعمق ولا تحتوي بصفة خاصة على مفاهيم ثورية مبتكرة. تُوَظِف حِزمة *ح ب ع ك* أُسلوب بنية المرة الواحدة التي لا يتكرر فيها استِخدام مِفتاح أكثر من مرة. تجمع الحِزمة بين العديد من خوارزميات التعمية القوية المعروفة التي تشمل خوارزميات المفتاح السري (المتماثِل) بالإضافة إلى خوارزميات المِفتاح العام (غير المتماثِل) مثل خوارزمية الجمل وخوارزمية آر إس إيه اللتين أثبت تحليلُ كشف المعمّى لهما صُمودهما النِسبي أمام أية هجماتٍ قوية. ووفقا لعدد الخوارزميات المستعملة، يُمكِن تحقيق مُستويات مُتفاوتة من الأمان بدءاً بالمستوى المنخفِض ومُروراً بالمستوى المتوسِط ووصولاً إلى المستوى العالي. يتم توليد مجموعة من المفاتيح عشوائيا بحيث يستخدم كل منها مرة واحدة فحسب في أية عملية للتعمية وكشف المعمى. وطبقا لحجم الرسالة المراد تعميتها، تستخدم هذه المفاتيح وفقا لأساليب وطرائق مختلفة يتحكم فيها المستخدم. تدعُم الحِزمة النُصوص العربية والأنجليزية ويُمكِن تحسينها لتشمل لغاتٍ أُخرى، كما يُمكِن استِعمالُها مع أنواعٍ مُختلِفة من المنصات دون ما حاجةٍ إلى عتاد مادي مُحدّد. وللحِزمة مقدِرة ذاتية على تحويل الصور إلى نصوص والعكس, الأمر الذي يُمثِل خُطوة رئيسة في استِغلال خصائص التعمية لإدراك أغراض التخفية. وبينما ترتكز طريقة الحزمة في التخفية أساسا على المناقلة بين الصور والنصوص فإن الحزمة يمكنها أيضا تخفية الصور في صور أخرى. تُراجِع الورقة الرياضيات الخاصة بالعمليّات المستخدمة، كما تَصِف قُدُرات الحِزمة المنفَذّة، وتُبين نفعها من خِلال أمثِلة توضيحية. كذلك تتشعب الورقة أحيانا لتلقي الضوء على إسهامات العرب خلال الحقبة الإسلامية الزاهرة في علم التعمية بوجه عام وفي كشف المعمى بوجه خاص.